Sécurisation d'un système de transactions sur terminaux mobiles

Chrystel Gaber

Université de Caen Basse-Normandie

École doctorale SIMEM

Thèse de doctorat

présentée et soutenue le : 24/10/2013

par

Chrystel Gaber

pour obtenir le

Doctorat de l'Université de Caen Basse-Normandie

Spécialité : Informatique et applications

Sécurisation d'un système de transactions sur terminaux mobiles

Directeur de thèse : Marc Pasquet

Co-directeur de thèse : Pascal Urien

Jury

AbdelMadjid Bouabdallah	Professeur des Universités à l'Université Technologique de Compiègne	(Rapporteur)
Isabelle Chrisment	Professeur des Universités à Télécom Nancy Université de Lorraine	(Rapporteur)
Samia Bouzefrane	Professeur au CNAM	
Ernesto Damiani	Professeur à l'Université de Milan	
Mohammed Achemlal	Ingénieur Orange	(Encadrant entreprise)
Marc Pasquet	Professeur des Universités à l'ENSICAEN	(Directeur de thèse)
Pascal Urien	Professeur Télécom ParisTech	(co-Directeur de thèse)

Pour Alexandre, Jacques et mes grand-parents

Remerciements

Ce travail de thèse s'est inscrit principalement dans le cadre du projet européen FP7 MASSIF, MAnagement of Security information and events in Service InFrastructure. Pour mener ces travaux à terme, j'ai bénéficié de l'appui d'un certain nombre de personnes à qui je souhaiterais témoigner toute ma gratitude.

En premier lieu, je remercie les membres du jury qui ont accepté d'évaluer la qualité de mon travail. Je remercie les rapporteurs de cette thèse Isabelle Chrisment, Professeure à TELECOM Nancy et Abdelmadjid Bouabdallah, Professeur à l'Université Technologique de Compiègne. Je leur suis reconnaissante pour l'intérêt qu'ils ont porté à mes travaux de recherche, tout en ayant un regard critique et juste. Je remercie Samia Bouzefrane, Maître de Conférences au CNAM et Ernesto Damiani, Professeur à l'Université de Milan d'avoir accepté de faire partie de mon jury de thèse.

J'exprime ma grande gratitude à Mohammed Achemlal de m'avoir acceptée en thèse au sein d'Orange et de m'avoir encadrée pendant ces trois années. Je le remercie également pour la confiance, le soutien et la sympathie qu'il m'a témoignés durant cette période.

Je remercie également mes directeurs de thèse Marc Pasquet et Pascal Urien pour leur encadrement ainsi que pour la confiance qu'ils m'ont accordée durant ces trois années.

J'exprime également toute ma reconnaissance à Baptiste Hemery pour les nombreuses discussions et brainstormings sans lesquels nombre de mes travaux de recherche n'auraient vu le jour. Ses relectures méticuleuses et son soutien indéfectible ont été une aide précieuse pour accomplir cette thèse.

Je remercie Jean-Claude Paillès pour m'avoir initiée à l'informatique de confiance et m'avoir donné envie de travailler dans ce domaine. Sa passion et ses enseignements sont autant d'inspirations pour moi.

Mes remerciements vont également à Romain Giot pour les longues discussions, marches ainsi que l'aide qu'il m'a apportée en tant que collègue dans le projet européen MASSIF.

Je remercie Yvan Rafflé pour sa relecture critique, constructive et extrêmement enrichissante de ma thèse.

Merci à Christophe Rosenberger pour m'avoir initiée à la recherche à travers mon stage de deuxième année d'école d'ingénieurs et pour m'avoir, en tant que directeur de l'équipe Monétique et Biométrie, donné l'opportunité de réaliser une thèse.

Je remercie également Anne Boutroux et Jean-François Misarsky pour leur accueil au sein de leur équipe au sein d'Orange.

Je tiens également à remercier l'ensemble des acteurs qui ont construit la monétique à Caen et sans qui l'équipe de recherche à laquelle j'ai appartenu ne serait pas ce qu'elle est : Félix Cuozzo, René Lozach, Marc Pasquet, Sylvain Vernois, Wilfried Aubry, Vincent Alimi, Marie Réveillhac, Wipa Chaisantikulwat, Olivier Catherine, Sébastien Duval, Joan Reynaud, Julien Mahier et Baptiste Hemery.

Je remercie aussi mes ami(e)s et qui m'ont aidée aussi bien dans le travail que dans la vie personnelle lorsque j'en avais besoin.

Merci à toi Mohamed pour ton aide à la fin de ma troisième année de thèse et pour les bons moments passés au cours de ces six derniers mois. Persévère, tu le trouveras un jour le gouvernement central XD. En tous les cas, je te souhaite à toi aussi une bonne continuation pour la thèse. De toute manière, on se recroisera pendant les trois prochaines années.

Merci Hachem pour ta présence et ton soutien pendant ces trois années de thèse commune. Tu as été un pilier et un modèle pour moi. Je te souhaite bon courage et bonne chance. La fin du tunnel est proche !

Sylvain, V. merci pour le coup de pouce, très apprécié.

Sylvain J., je te remercie pour ta bonne humeur, ta folie entraînante. Tu m'as littéralement donné le sourire. Avec toi tout est legen - wait for it - dary :)

Merci Kahina, j'ai trouvé en toi plus qu'une amie. Nous avons partagé un certain nombre de peines, d'inquiétudes et surtout de fous rires. Pour tous ces moments complices, je te remercie. Je te souhaite bonne chance pour la suite de ta thèse.

Johann, merci d'avoir été le grand frère protecteur et rassurant. Merci de m'avoir aidée à aller de l'avant et à prendre de la hauteur. Je te remercie pour ton soutien, ton écoute patiente et les nombreux moments de joie partagés. Maintenant, que je suis docteure aussi, c'est parti pour la conquête du monde ! Merci aussi de m'avoir fait découvrir GOT ! Vallhar Morghullis. Bon, c'est bien beau tout ça mais c'est quand le prochain week end jeux/ripaille/films où on dépose le cerveau ?

Je tiens également à te remercier Annick, tu es la meilleure des belle-mères !

Baptiste, c'est un bonheur de partager ta vie. Je te remercie pour ton soutien inconditionnel et ta patience. Merci d'avoir été là pour écarter les doutes, soigner les blessures et partager les joies. Cette thèse est aussi la tienne.

Voilà, je suis arrivée au bout et mes derniers remerciements vont pour vous, mes parents. Sans vous, ce parcours n'aurait pas été possible, je ne vous remercierai jamais assez.

So long, and thanks for all the fish.
The Hitchhiker's Guide to the Galaxy
Douglas Adams. 1979.

Résumé

Les transactions sur mobile suscitent depuis quelques années un intérêt grandissant. Différents usages et modèles de transactions sur mobile existent. Il peut s'agir d'une forme de paiement supplémentaire, comme dans les pays où le taux de bancarisation est élevé ou d'une forme de paiement nouvelle qui transforme les usages, comme dans les pays faiblement bancarisés. Ce dernier cas représente une opportunité pour les opérateurs de téléphonie mobile. Ceux-ci peuvent proposer des services dans un marché qui n'est pas encore investi par les banques et où le taux de pénétration des téléphones mobiles est élevé. Cette thèse se place dans le cadre de la sécurisation d'un service de transactions sur terminaux mobiles géré par un opérateur de téléphonie mobile. Les systèmes étudiés dans cette thèse se base sont représentés par un modèle de paiement dit à trois coins contrairement aux modèles bancaires traditionnels dis à quatre coins. Dans ce modèle de paiement particulier, les utilisateurs souscrivent au service offert par l'opérateur de téléphonie mobile et réalisent les transactions avec de la monnaie électronique émise par l'opérateur et qui peut être acquise auprès d'agents de distribution. Il s'agit d'un circuit fermé où les transactions ne peuvent être réalisées qu'entre acteurs ayant des accords avec l'opérateur ou souscrivant également au service et en utilisant exclusivement la monnaie électronique.

La sécurité des messages liés aux transactions s'appuie sur celle offerte nativement par les canaux de signalisation des réseaux mobiles. La sécurité n'est pas assurée de bout-en-bout entre l'application de paiement et la plateforme de paiement. Cela pose problème dans le cadre des futurs réseaux de données de quatrième génération car ceux-ci n'incluent pas les canaux de signalisation utilisés aujourd'hui pour transporter les instructions de paiement. De plus, dans la majorité des cas, les données ne sont chiffrées qu'entre le terminal mobile et la première antenne du réseau. Ensuite, elles transitent en clair. Dans ce modèle, on prend l'hypothèse que le cœur du réseau est de confiance puisqu'il est maîtrisé par l'opérateur. Cependant, les données peuvent

transiter par le réseau d'un opérateur concurrent dans le cadre de l'itinérance ou par un réseau filaire tout-IP dans le cadre des réseaux de quatrième génération. Dans ces contextes, les instructions de paiement peuvent transiter en clair par des réseaux non maîtrisés par l'opérateur qui propose le service de transactions sur mobile. À ces limitations liées au réseau, s'ajoute le fait que les *smartphones*, téléphones intelligents, sont vulnérables aux virus. Ceux-ci peuvent notamment créer des transactions à l'insu de l'utilisateur.

Cette thèse concerne la sécurisation de ces services. Nous proposons tout d'abord des protocoles permettant d'assurer la sécurité de bout-en-bout entre l'application de paiement dans la carte SIM et la plateforme de paiement de l'opérateur. La carte SIM est un élément sécurisé maîtrisé par les opérateurs de téléphonie mobile. A notre connaissance, il n'existe pas de protocole de paiement permettant de répondre à nos problématiques et adapté à notre contexte. En particulier, la norme *EMV*, Europay Mastercard Visa, est un standard adapté aux paiements de proximité réalisés entre une carte à puce et un terminal de paiement électronique auprès d'un marchand. Il ne s'agit donc pas d'un standard pour les transactions réalisées entre deux terminaux mobiles. De plus, le modèle de paiement sur lequel se base EMV et celui considéré par la thèse reposent sur des contextes de risque différents.

L'adaptation des algorithmes de détection de fraudes aux services de transactions sur mobile est également un thème abordé dans la thèse. Une des problématiques de ce domaine de recherche est l'absence de données réelles provenant du terrain. Un simulateur permettant de modéliser les habitudes des utilisateurs légitimes et des parcours fraudeurs a été réalisé. De nombreuses méthodes de détection de fraudes existent et ont été appliquées au domaine des paiements par carte bancaire. Cependant, à notre connaissance, il n'existe pas d'étude similaire à celle présentée dans la thèse concernant l'adaptation de méthodes de détection de fraudes aux systèmes de transactions sur mobile correspondant à notre contexte. L'originalité de cette contribution réside dans le fait qu'un jeu de données provenant d'un service déployé a été utilisé pour créer le simulateur de données utilisé dans la phase d'étude des algorithmes de détection.

Summary

Mobile-based transactions have driven growing attention for the past few years. Various uses and types of mobile-based transactions exist. For example, in countries where banking services penetration rates are high, mobile payments are an additional means to carry out transactions. On the contrary, in countries with low banking penetration rates, they represent a new payment means which transforms behaviours. The latter case is an opportunity for mobile network operators as it enables them to propose services in a market with a high mobile penetration rate and very few banking actors. This work's framework is a mobile-based transaction system managed by a mobile network operator. This system follows a three-party payment system in contrast with traditional banking models which follow a four-party payment system. In this particular three-party payment system, users subscribe to the mobile network operator's payment service. They carry out transactions with the electronic money which is emitted by the operator and distributed by distributor agents. It is a closed-loop system where transactions can only be carried out with electronic money and with other subsribers.

The security of this type of service is based on the one natively provided by the mobile network signalling channels. End-to-end security between the payment application and the payment platform is not achieved. As a consequence, problems may arise in the fourth generation networks as they do not include the channels used today to transmit the payment instructions. Moreover, in most cases, the data are ciphered only between the mobile device and the network's first antenna. Afterwards, they are transmitted unciphered. This model is based on the assumption that the network's core can be trusted as it is managed by the operator. However, the data can be transported in a competitor's network in the context of roaming or in an all-IP wired network in the context of fourth generation networks. Under these circumstances, it is possible for a payment instruction to be transmitted unencrypted

in a network which is not managed by the operator. In addition, smarthones are vulnerable to viruses. The latter could in particular carry out transactions without the content of the user.

This thesis is about securing mobile-based transaction services. First of all, an architecture and several protocols are proposed to achieve end-to-end security between the application in the SIM card and the operator's payment platform. The SIM card in considered as a Secure Element which is managed by mobile network operators. To our knowledge, no payment protocole addresses the problems stated here and is adapted to this context. In particular, the EMV (Europay Mastercard Visa) standard is adapted to proximity payments carried out by a smartcard and an electronic payment terminal at a point of sale. It is therefore not adapted for transactions, whether transfers or purchases, carried out by two mobile devices. Moreover, the payment model on which EMV is based and the one considered in this thesis rely on two different contexts of risk.

The adaptation of fraud detection algorithms to mobile-based transaction services is the second topic addressed in this thesis. One issue in this field is the lack of data collected on the field. A simulator has been created in this thesis to model the behaviour of fraudulent and non-fraudulent users. Several fraud detection methods exist and were applied in the field of card transactions. However, to our knowledge, the study performed in this thesis to adapt fraud detection methods to this specific context of mobile-based transaction system is the first of its kind. The novelty of this thesis is that a database of transactions from an existing service was used to create the simulator on which the the study of the detection algorithms is based.

Table des matières

Introduction

Contexte de réalisation de la thèse

Le projet européen FP7 MASSIF [1] a pour objectif d'adapter les outils de gestion des événements et informations de sécurité, *SIEM* (Security Information and Event Security) aux problèmes de sécurité de niveau applicatif. Actuellement, ces outils s'appliquent à la surveillance d'attaques sur les réseaux comme des intrusions par exemple. Orange, entreprise au sein de laquelle cette thèse est réalisée, propose dans ce projet un cas d'usage lié à la gestion des problèmes de sécurité pouvant viser les services de transactions sur terminaux mobiles.

Ces derniers représentent un secteur économique grandissant dans le monde. Par exemple, le service M-Pesa, mis en place en 2007 au Kenya, affichait en décembre 2011 environ 19 millions de souscrivants ce qui représente 70% des abonnés mobile au Kenya [CCK12]. De même, le service Orange Money est déployé dans 10 pays et regroupe 14% des abonnés mobile de ces pays [Ora12].

Ces deux services, Orange Money et M-Pesa, sont deux exemples de services de transactions sur mobile gérés par un opérateur. Contrairement au système bancaire, ce type de service fonctionne en cercle fermé. Les transactions ne peuvent être réalisées qu'entre les membres de ce cercle. Les souscrivants de ce type de service utilisent de la monnaie électronique qui est émise par l'opérateur téléphonique sous certaines conditions et qui ne peut être utilisée qu'entre les souscrivants.

Problématique et objectifs

Comme tous les services financiers, les systèmes de transaction sur terminaux mobiles sont soumis aux risques de fraudes. Les enjeux économiques, stratégiques,

1. http://www.massif-project.eu/

judiciaires et politiques sont considérables. En effet, le succès de tels services de paiement se base essentiellement sur la confiance que les utilisateurs ont dans le service. Un taux élevé de fraude porterait atteinte à celle-ci. De plus, selon la nature et l'impact de la fraude, celle-ci peut mettre en cause l'existence même du service ou avoir des conséquences sur l'économie de la zone où il est déployé.

De plus, il peut exister une obligation de détection et de suivi des fraudes. Par exemple, en France, l'article L561-15 du Code monétaire et financier soumet les banques, établissements de paiement et établissements de monnaie électronique à une obligation de vigilance. Plus particulièrement, ces établissements doivent déclarer les opérations ou les sommes dont ils savent *qu'elles proviennent d'une infraction passible d'une peine privative de liberté supérieure à un an ou participent au financement du terrorisme*. Selon l'article L515-6 du Code monétaire et financier, ces établissements doivent procéder à un *examen attentif des opérations effectuées en veillant à ce qu'elles soient cohérentes avec la connaissance actualisée qu'elles ont de leur client*. En conséquence de ces articles, toute activité suspecte est rapportée au gouvernement français via le système Tracfin [2]. Il existe des actions en justice suite à des fraudes contre des banques [3]. De plus, les accords de Bâle II [Nou06], applicables aux banques, exigent que les risques opérationnels, dont le risque de fraude fait partie, soient couverts par les fonds propres.

La gestion du risque de fraude est donc une problématique importante dans tout service. Cela est plus particulièrement vrai dans le cadre des paiements sur mobile puisqu'ils ont été déployés relativement récemment et disposent donc d'un retour d'expérience limité. En effet, les systèmes de transaction sur mobile ont commencé à se développer depuis une dizaine d'années. Ils ont connu un réel essor avec le lancement et le succès de M-Pesa en 2007. Par contre, les systèmes de transaction sur carte bancaire existent depuis une trentaine d'années. Cette thèse se place dans le cadre de la sécurisation des services de transactions sur mobiles décrits ci-dessus. L'objectif est d'étudier cette problématique sous deux angles complémentaires : la prévention et la détection de ces transactions frauduleuses. Le premier axe cible les solutions techniques permettant d'empêcher les attaquants de réaliser la fraude. Le second axe concerne les méthodes permettant de déceler des fraudes.

2. http://www.economie.gouv.fr/tracfin/accueil-tracfin
3. http://www.lefigaro.fr/conso/2012/09/25/05007-20120925ARTFIG00535-jusqu-o-les-banques-doivent-elles-surveiller-vos-comptes.php

Contributions

Les contributions de cette thèse concernent les deux problématiques de la sécurisation décrites ci-dessus. En terme de prévention, une architecture et différents protocoles sont proposés en se plaçant dans un contexte où l'utilisation des réseaux de quatrième génération tout-IP est répandue. Les propositions permettent de sécuriser les transactions de bout-en-bout entre l'application dans le terminal mobile et la plateforme de paiement. Différents modes liés à la connectivité entre les différents acteurs sont considérés. Ainsi, des transactions pourront être réalisées si tous ou une partie des acteurs de la transaction sont connectés à un réseau qu'il s'agisse d'une architecture déployée par l'opérateur ou une architecture déployée par un tiers qui n'est pas nécessairement de confiance. Une approche permettant d'effectuer des transactions en mode déconnecté est également proposée.

La deuxième contribution de la thèse concerne la détection de la fraude. Celle-ci est indépendante des technologies et de la typologie de réseau utilisée. Elle peut s'adapter aux systèmes de transaction terminaux mobiles actuels. L'objectif est d'étudier l'application d'algorithmes de classification au problème de détection de fraudes dans les systèmes de transactions sur mobile. De par la rareté des données publiques et la mise en place relativement récente des services considérés ici, il n'existe pas de base de données de transactions exploitables pour la détection de fraude. En effet, ceux-ci ne contiennent pas de transactions labellisées permettant de contrôler l'efficacité des algorithmes de détection. L'originalité de cette thèse réside dans l'utilisation d'une base de données provenant d'un système déployé sur le terrain. Cependant, celle-ci est confidentielle et ne comporte pas de vérité terrain permettant d'évaluer l'efficacité des algorithmes de classification. Elle a donc été utilisée pour créer un simulateur de données afin de pallier ce problème de manque de données. Les données ainsi produites sont utilisées pour réaliser l'étude sur l'adaptation de méthodes de classification.

Organisation du manuscrit

Ce manuscrit est composé de cinq parties. Tout d'abord, le contexte de la thèse est précisé et différentes notions liées au paiement mobile et à la gestion de la fraude sont définies. Ensuite, un état de l'art est réalisé sur la sécurisation de tels services. Celui-ci s'articule autour des deux axes de la thèse concernant la prévention et la détection. Plus particulièrement, les architectures sécurisées de systèmes de transactions sur mobile et les algorithmes de classification ont été explorés et détaillés. Les troisième et quatrième chapitres détaillent les contributions de la thèse. Le troisième chapitre

présente les contributions concernant l'architecture du système de transactions sur mobile proposé dans le cadre de cette thèse. Le quatrième chapitre détaille l'étude sur l'adaptation des méthodes de classification et la création d'un générateur de données articifielles nécessaire à celle-ci. Enfin les conclusions de cette thèse sont rendues et les perspectives de ces travaux sont présentées.

Chapitre 1

Le paiement mobile et la gestion de la fraude

L'objectif de ce chapitre est de spécifier les systèmes et services sur lesquels se basent les travaux de cette thèse. Pour cela, nous définissons tout d'abord différentes notions liées à l'écosystème des transactions réalisées sur un terminal mobile. Ensuite, nous utilisons ces notions pour positionner les systèmes considérés cette thèse dans cet écosystème. Les notions de fraude et gestion de la fraude sont également définies.

Sommaire

1.1 L'écosystème du paiement mobile

Le paiement mobile peut désigner plusieurs types de services. L'objet de cette partie est de définir l'écosystème et le contexte dans lequel se place cette thèse. Pour cela, nous définissons les notions de paiement et de monnaie sur lesquelles se base cette thèse. Les institutions pouvant être impliquées dans le paiement sont également présentées. Finalement la notion de paiement mobile est spécifiée en utilisée

les définitions précédentes. Plusieures formes de paiement mobile sont également présentées.

1.1.1 L'opération de paiement

Dans le cadre de cette thèse, nous retenons la définition juridique du paiement [Répa, Wér07]. D'après l'article 1234 du Code Civil, le paiement est un des moyens qui permettent d'éteindre une obligation définie par une convention qui lie une ou plusieurs personnes (article 1101 du Code Civil).

Nous considérons ici les conventions au sens large. Celles-ci peuvent être explicites par la signature d'un contrat. C'est par exemple le cas d'un bail qui oblige un locataire à régler son loyer au propriétaire du logement occupé. La convention peut également être tacite. C'est par exemple le cas d'une vente : le vendeur s'engage à donner la marchandise en l'échange d'argent. Cela correspond également au cas où un migrant choisit de soutenir financièrement sa famille restée au pays : cette personne s'est engagée moralement à le faire.

Dans le langage courant, le paiement correspond également à une opération consistant à acquérir un bien ou un service grâce à une contrepartie. Pour éviter la confusion avec la définition que nous avons donnée précédemment, dans la suite de ce manuscrit, nous désignons cette opération par les termes achat ou paiement marchand.

Enfin le retrait et de débit sont des opérations liées à la gestion de compte. Il ne s'agit pas de paiement à proprement parler mais nous considérons que c'est le cas dans le cadre de la thèse.

Le paiement est réalisé grâce à une somme d'argent qui est transférée à l'aide d'un moyen de paiement. La loi bancaire du 24 janvier 1984 [Répb] définit les moyens de paiement comme étant *tous les instruments qui, quel que soit le support ou le procédé technique utilisé, permettent à toute personne de transférer des fonds*. Les pièces et les billets sont des moyens de paiement ainsi que les chèques, les virements, ou la carte bancaire.

Pour qu'un paiement puisse avoir lieu, une certaine organisation et intermédiation est nécessaire. En effet, l'argent échangé doit être émis par une entité et des moyens de paiement doivent être mis à disposition des utilisateurs. Certaines procédures de paie-

ment nécessitent l'intervention d'intermédiaires et, donc, de systèmes de *règlement*[1] et de *compensation*[2]. Par exemple, la gestion de la monnaie scripturale nécessite deux intermédiaires bancaires ainsi qu'un système de *règlement* et de *compensation* qui permet l'*interbancarité*[3]. L'ensemble de ces instruments, procédures, établissements constitue un *système de paiement* [Com03].

De manière générale, les quatre fonctions rencontrées dans un système de paiement sont l'*émission* qui concerne la fabrication et la distribution d'un instrument de paiement ; l'*acceptation* qui concerne la réalisation d'une transaction ; l'*acquisition* qui correspond à la collecte des transactions auprès des accepteurs et la *compensation* des opérations entre les différentes parties du système.

Les différents acteurs mis en jeu sont le *porteur* qui dispose d'un instrument de paiement ; l'*accepteur* qui accepte l'instrument de paiement du porteur pour réaliser une opération ; l'*émetteur* qui fournit l'instrument de paiement au porteur et l'*acquéreur* qui interagit avec l'accepteur pour gérer les transactions. Généralement, dans le domaine des paiements par carte à puce, l'accepteur est un marchand aurpès duquel un achat est réalisé. Dans le cadre de cette thèse, nous considérons que l'accepteur peut à la fois être un marchand ou un particulier qui possède également un instrument de paiement.

Les systèmes de paiement suivent généralement l'un des deux modèles de référence représentés en figures 1.1 et 1.2. Le modèle *quatre-coins* correspond à un modèle où deux entités différentes gèrent les fonctions d'émission et d'acquisition. Dans le modèle *trois coins*, une seule entité gère ces deux fonctions. Ce modèle implique que les transactions sont réalisées dans un cercle fermé. Tous les acteurs pouvant réaliser des transactions souscrivent à un système particulier. Ils disposent d'un moyen de paiement qui est valable uniquement dans un réseau propre à ce système. Des exemples typiques de ce type d'architecture sont American Express[4], Diner's

1. Règlement. Lien entre des transferts de titres et des transferts de fonds permettant de s'assurer que la livraison d'un actif ne s'effectue que si le paiement est réalisé et vice-versa http://www.fbf.fr/fr/mieux-connaitre-la-banque/lexique-R

2. Compensation. Opération journalière par laquelle les banques soldent leurs créances et dettes réciproques http://www.fbf.fr/fr/mieux-connaitre-la-banque/lexique-C

3. Interbancarité. Réseau de relation entre les banques qui permet notamment la fourniture de services par une banque aux clients d'autres banques http://www.fbf.fr/fr/mieux-connaitre-la-banque/lexique-I

4. https://www.americanexpress.com/france/

Club [5] et Chèque Déjeuner [6].

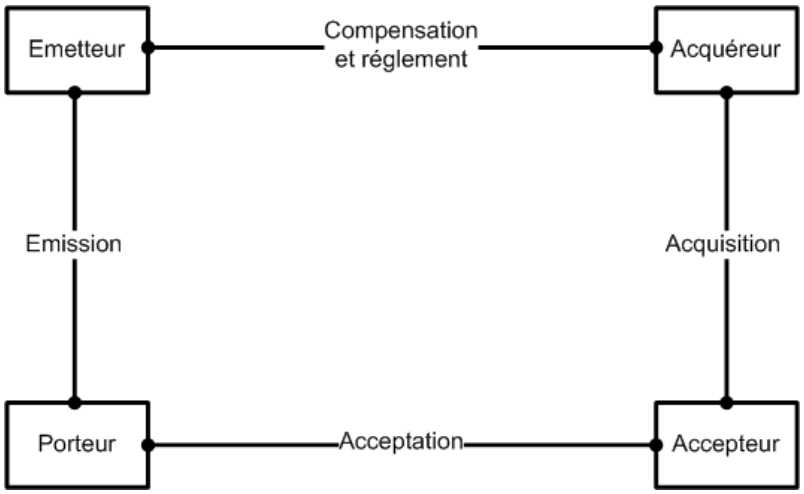


FIGURE 1.1 – Modèle de référence d'un système quatre coins

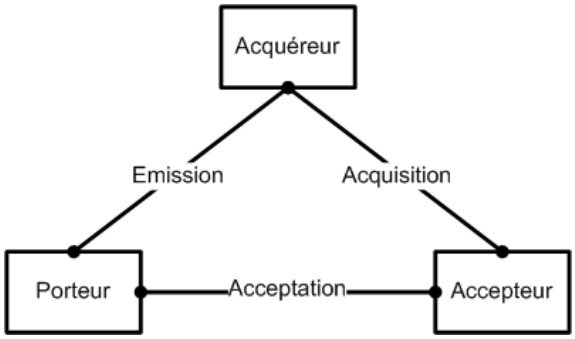


FIGURE 1.2 – Modèle de référence d'un système trois coins

Le paiement électronique est apparu avec la dématérialisation de la monnaie et les avancées technologiques. Selon Wéry [Wér07], le paiement électronique regroupe l'ensemble des moyens de paiement initialisés avec un système qui a recours à l'électronique. Cette définition regroupe aussi bien les paiements qui se font via internet ou les téléphones mobiles que les paiements qui se font avec une carte bancaire.

Classification usuelle des paiements

Les paiements peuvent être classifiés selon leur valeur, la localisation des acteurs, le le statut de ceux-ci et le moment où le compte du payeur est débité. En se ba-

5. http://www.dinersclub.fr/
6. http://www.cheque-dejeuner.com/

sant sur le facteur de localisation, nous distinguons deux formes de paiement. La première correspond au paiement de proximité où le payeur et le payé partagent leur localisation. Cela concerne par exemple les achats en face à face chez un marchand ou les retraits sur un distributeur. La deuxième forme de paiement correspond aux paiements à distance où les deux acteurs ne se trouvent pas sur le même lieu. Cela correspond aux transferts d'argent entre particuliers, l'achat de marchandises sur internet ou le paiement de factures [7].

En considérant la somme du paiement, nous pouvons distinguer le micropaiement, généralement limité à une valeur inférieure à 10 euros, et le macropaiement, généralement supérieur à 10 euros. Cette limite est relative et dépend du contexte.

On distingue quatre formes de paiements selon le statut du payeur et du payé. Cette distinction correspond souvent à une différents segments de marché pouvant être adressés par un service de paiement. Ces différents segments peuvent correspoindre à des typologies (paiement distant ou de proximité, macro- ou micro-paiement) qui varient. Entre autres, les catégories suivantes peuvent être citées :

- C2C : Client to Client. Il s'agit des transferts entre particuliers ;
- C2B : Client to Business. Il s'agit de paiements de particuliers à entreprises. Il peut s'agir de paiements de marchandises tout comme le paiement de taxes au gouvernement ;
- B2C : Business to Client. Il s'agit de paiements d'entreprises à particuliers comme des remboursements ou le paiement de prestations sociales ;
- B2B : Business to Business. Il s'agit de paiements entre entreprises.

Le paiement peut être prépayé, post-payé ou immédiat [Kar04]. Un achat dont le paiement est reporté sur la facture téléphonique du client est post-payé. La carte à débit immédiat est un exemple typique de paiement immédiat. L'utilisation d'un compte, électronique ou virtuel, qui contient des valeurs préalablement achetées correspond à du paiement prépayé. C'est le cas de Moneo [mon] par exemple.

Le paiement permet d'échanger des biens et des services. Ainsi, le troc est considéré comme la première forme de paiement. Cependant, la valeur des objets vendus est difficilement comparable et une valeur de référence est nécessaire. Le troc a donc évolué vers l'utilisation de la monnaie comme moyen d'échange et de valeur de

7. Selon les pays, les abonnés n'ont d'autre choix que de se déplacer dans une agence qui peut être très éloignée de leur lieu de vie. L'intérêt du paiement mobile est dans ce cas est de permettre le paiement de factures à distance et d'éviter le développement d'agences de facturiers.

référence. La monnaie et ses fonctions sont présentées dans la partie suivante.

1.1.2 La monnaie

Nous avons vu précédemment que la monnaie est une composante essentielle du paiement. Les différentes fonctions et formes de la monnaie sont maintenant précisées.

La monnaie a trois fonctions essentielles, outre le pouvoir d'éteindre une créance [DGKN97, Lan99]. Une monnaie est une *unité de compte*, elle permet d'évaluer une valeur et sert de référence dans un système monétaire. Cette fonction permet d'avoir une définition égalitaire et juste de la valeur des marchandises. La monnaie est aussi une *réserve de valeur*, elle peut être conservée entre le moment où elle a été acquise et le moment où elle va être utilisée pour un achat. Selon Keynes [Key36], l'importance de la monnaie découle essentiellement du fait qu'elle constitue un lien entre le présent et l'avenir. Une monnaie est un *instrument d'échange*, elle facilite et accélère les échanges. Elle permet d'acquérir des biens et peut être transférée. On peut également considérer que la monnaie a une quatrième fonction, celle d'instrument de politique économique qui permet à l'État d'influencer l'activité économique, les objectifs de croissance et la stabilité des prix [8].

Aujourd'hui, la monnaie n'a pas de valeur intrinsèque alors qu'elle en avait une lorsque les pièces étaient fabriquées dans un métal précieux comme l'or ou l'argent. Sa valeur est donc basée sur son acceptabilité et la confiance que les différents acteurs économiques ont en elle. Actuellement, cette confiance se base sur celle de l'État émetteur de la monnaie. En effet, la gestion de la monnaie est une fonction régalienne visant à assurer la souveraineté économique du pays. L'État peut, entre autres, donner plus de poids à une monnaie particulière en lui conférant un *cours légal*. Dans ce cas, une loi interdit à toute personne de refuser de recevoir la monnaie qui bénéficie du cours légal en règlement d'une dette dans l'unité monétaire du pays. En France, le Code monétaire et financier fixe l'emploi des différentes formes de monnaie. De manière générale, seule la monnaie fiduciaire, définie ci-dessous, bénéficie du cours légal. Ces dispositions ne sont pas communes ailleurs dans le monde. Il existe par exemple des zones où la monnaie électronique a cours légal : c'est le cas de Hong Kong [BS03]. Il existe aussi d'autres zones où la monnaie fiduciaire n'a pas cours légal. Par exemple, en Ecosse et en Irlande du Nord, aucune monnaie n'a le statut de cours légal [Ban]. Ces exemples montrent que le cours légal n'est pas le

8. http ://www.senat.fr/eco/ec-04/ec-040.html

seul élément permettant d'influencer l'acceptabilité de la monnaie.

Les différents types de monnaie peuvent être différenciés en fonction de leur forme et de leur émetteur. En se basant sur le type de support, on peut distinguer trois sortes de monnaie. Il existe tout d'abord la monnaie fiduciaire qui correspond aux pièces et aux billets, puis la monnaie scripturale qui correspond à des soldes inscrits sur un compte bancaire. La dernière forme est la monnaie électronique. Il s'agit d'une valeur stockée sur un support électronique indépendant d'un compte bancaire. On parle de porte-monnaie électronique si la monnaie électronique est stockée localement sur un appareil électronique autonome. On parle de porte-monnaie virtuel si la monnaie électronique est stockée sur un serveur qu'un appareil électronique permet d'accéder.

La monnaie électronique peut prendre plusieurs formes. On peut la trouver sous la forme de jetons qui correspondent à une certaine valeur. Dans ce cas, la somme d'argent détenue dans le porte-monnaie électronique correspond à la somme des valeurs des jetons. La monnaie électronique peut aussi être une valeur numérique stockée dans le porte-monnaie électronique. Cette forme ressemble au solde d'un compte. Enfin, il existe une représentation hybride de la monnaie électronique. La monnaie est stockée en tant que valeur numérique sur le support mais l'échange prend la forme d'un jeton. Généralement, le porte-monnaie électronique ou virtuel est fermé. C'est à dire que le marchand ne peut pas réutiliser directement la monnaie électronique qui lui est versée. Cette monnaie doit passer sur un compte bancaire et le paiement est enregistré à ce moment. Ce n'est pas le cas du porte-monnaie ouvert. Cette forme n'est pas utilisée car elle pourrait générer des problèmes économiques liés à la circulation de la masse monétaire [AS02, CG96].

Dans la suite de ce manuscrit, les termes *comptes prépayés virtuels* et *comptes prépayés électroniques* seront privilégiés à ceux de *porte-monnaie électronique* et de *porte-monnaie virtuel*. En effet, le terme *porte-monnaie* est utilisé dans plusieurs offres commerciales très différentes et peut donc porter à confusion. De plus, l'appellation *compte prépayé* permet de mieux mettre en évidence le fait que la monnaie manipulée dans ce type de système correspond à des unités de valeurs achetées au préalable.

Certains auteurs comme Perdrix [Per94], Bounie et Soriano [BS03] reconnaissent à la monnaie électronique le statut de monnaie alors que des auteurs comme Lanskoy [Lan99], Simon [Sim99] réfutent cette thèse et désignent la monnaie électronique

comme étant simplement un nouveau moyen de paiement comme le chèque. Dans la suite de ce manuscrit nous allons nous baser sur [Per94, DGKN97, Ban10, AS02, Bas98, Eur98] qui considèrent que la monnaie électronique n'est assimilable ni à la monnaie scripturale ni à la monnaie fiduciaire et constitue donc une nouvelle forme de monnaie. Une définition précise de la monnaie électronique est donnée par la directive européenne 2000/46/CE [Uni00] et sa remplaçante la directive 2009/110/CE [Uni09]. En se basant sur ces deux textes, la monnaie électronique est définie comme étant :

1. *Une valeur monétaire stockée sous forme électronique ;*
2. *représentant une créance sur l'émetteur ;*
3. *remise en échange de fonds d'un montant dont la valeur n'est pas inférieure à la valeur monétaire émise ;*
4. *utilisée pour faire des opérations de paiement (versement, transfert ou retrait des fonds) ;*
5. *acceptée par une personne physique autre que l'émetteur de monnaie électronique.*

En se basant sur l'émetteur, on distingue également trois sortes de monnaie. La première correspond à la monnaie centrale. Il s'agit de la monnaie émise par la banque centrale d'un pays. Une partie de la monnaie centrale est en circulation sous la forme de pièces et de billets. L'autre partie correspond aux comptes que les différentes banques privées ont auprès des banques centrales. La seconde est la monnaie bancaire. C'est la monnaie émise par les banques commerciales. Finalement, la monnaie privative est celle émise par des commerçants ou prestataires de services et plus généralement par un organisme n'ayant pas un statut bancaire. Cette monnaie n'est utilisable que dans un réseau restreint. Il s'agit par exemple de la monnaie utilisée dans des réseaux virtuels de jeux en ligne comme Facebook, World Of Warcraft ou Diablo 3, des Systèmes d'Échange Libre, ou des réseaux proposant des services prépayés. Le statut, la valeur de ce type de monnaie dépend des contrats commerciaux et accords passés entre les membres du réseau et le fournisseur de service. Elle dépend également de la relation entre le fournisseur de service, le régulateur et la banque centrale des pays concernés.

Le schéma en figure 1.3, montre comment se recoupent ces différentes formes de monnaies. Les pièces et les billets sont uniquement émis par la banque centrale. La monnaie fiduciaire est donc incluse dans la monnaie centrale. Une partie de la monnaie centrale se trouve sous la forme de comptes de banques commerciales

détenus à la banque centrale. La monnaie centrale n'est donc pas restreinte à la monnaie fiduciaire et s'élargit à la monnaie scripturale. La monnaie bancaire peut prendre la forme de dépôt à vue (monnaie scripturale) ou de monnaie électronique (Monéo par exemple). Il peut également exister de la monnaie électronique privative, il s'agit par exemple de la monnaie émise dans des services tels que M-Pesa au Kenya. Notons que la monnaie électronique bancaire est parfois assimilée à de la monnaie scripturale bien qu'il y ait des différences entre ces deux formes de monnaie [Eur98].

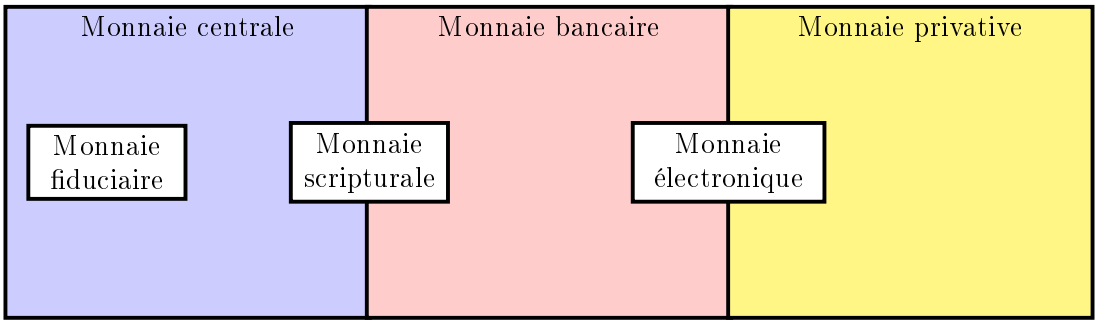


FIGURE 1.3 – Les monnaies

Nous avons vu ci-dessus qu'il existe différents acteurs pouvant émettre différentes sortes de monnaie. Dans la partie suivante, nous allons approfondir cette étude en nous intéressant aux différentes institutions pouvant créer de la monnaie.

1.1.3 Les institutions

Les institutions financières monétaires sont des institutions qui peuvent créer de la monnaie. Les institutions financières non monétaires quant à elles n'ont pas cette capacité. Dans la zone Euro, les institutions financières monétaires comprennent la banque centrale européenne, les banques centrales nationales, les établissements de crédit et les autres institutions dont l'activité est de recevoir des dépôts de la part d'agents non financiers. Dans cette partie, nous allons nous intéresser plus particulièrement aux principaux acteurs qui peuvent intervenir dans le paiement mobile. Il s'agit des banques centrales, des établissements de crédit, des établissements de monnaie électronique. Nous nous intéresserons également au concept de non-banque.

La banque centrale

La banque centrale est l'institution qui décide de la politique monétaire d'un pays. Les différentes missions d'une banque centrale peuvent changer selon les pays

mais elles consistent généralement à :

- Emettre la monnaie fiduciaire ;
- Veiller à la stabilité de la monnaie ;
- Veiller à la stabilité du système de paiement ;
- Contrôler l'activité des banques ;
- Conseiller les pouvoirs publics sur le système de paiement national et sa sécurité.

Les banques ou établissements de crédit

L'activité des banques est définie par la loi du 24 janvier 1984 [Répb] en France, et par la directive 2006/48/CE [Eur06]. Selon ces règles, les activités qui constituent le cœur de métier d'une banque consistent à :

- Recevoir des fonds du public ;
- Accorder des crédits ;
- Gérer des moyens de paiement.

Les banques sont considérées comme des instituts financiers monétaires car elles créent de la monnaie scripturale en accordant des crédits. Ces règles spécifient aussi que toute banque doit avoir un agrément pour pouvoir exercer. En France, celui-ci est délivré par le Comité des Établissements de Crédit et des Entreprises d'Investissement. Les banques sont soumises à des dispositions prudentielles spécifiques. Celles-ci fixent les obligations que les banques doivent respecter pour se prémunir face à certains risques. Le titre 5 de la directive 2006/48/CE [Eur06] fixe les principes de la surveillance prudentielle des banques. En France, ces dispositions sont définies dans la section 7 du code monétaire et financier. Les accords de Bâle réglementent l'activité bancaire et en particulier le risque de crédit défini en section 1.3.1.

Les non-banques

La notion de non-banque n'existe pas en France car la loi ne permet pas l'exercice d'activités bancaires sans agrément [DGKN97]. Il existe cependant des pays qui permettent à des organismes non-bancaires qui n'ont pas de licence ni d'agrément d'exercer une activité bancaire. Une non-banque [CP02] est un organisme qui fournit des services bancaires sans être soumis aux mêmes dispositions prudentielles que les banques. Généralement, leur activité est restreinte. Typiquement, elles ne font pas partie du système de paiement et ne bénéficient donc pas de l'interbancarité. Souvent, elles n'ont pas non plus le droit d'accepter de dépôts du public.

Les établissements de monnaie électronique

Selon la directive européenne 2009/110/CE [Uni09], la monnaie électronique peut être émise par des entreprises n'ayant pas le statut d'établissement de crédit. Ces entreprises doivent cependant avoir un statut d'établissement de monnaie électronique. Elles ne peuvent pas recevoir de fonds du public ni proposer de crédit basé sur des fonds reçus du public. Elles peuvent simplement distribuer la monnaie électronique. Les fonds reçus dans ce cadre ne sont pas considérés comme des dépôts reçus du public car ils doivent être échangés sans délai contre de la monnaie électronique. Les fonds reçus doivent être protégés par les établissements de monnaie électroniques selon les dispositions de l'article 9 de la directive 2007/64/CE [Uni07].

Les conditions encadrant l'émission de monnaie électronique dépendent de la réglementation de chaque pays. En France, elles sont plus strictes que dans les autres pays européens puisque seuls les établissements de crédit peuvent émettre de la monnaie électronique [Fra02]. Ailleurs dans le monde, certains pays autorisent des non-banques à émettre de la monnaie électronique sans statut particulier à part des accords avec les banques centrales. Cela soulève cependant de nombreuses questions, notamment sur la protection des fonds des utilisateurs et sur la menace de concurrence avec l'activité des banques centrales [Eur98, Mes01].

Après avoir examiné le paiement en général, nous définissons le paiement mobile.

1.1.4 Le paiement mobile

Le commerce électronique a démarré dans les années 1980 avec le minitel. Ces transactions électroniques étaient les prémisses des services proposés aujourd'hui sur internet. Le paiement mobile s'inscrit dans la lignée des évolutions qui ont marqué la dématérialisation de la monnaie et la modernisation des moyens de paiement. Dahlberg *et coll.* [DMOZ08] et Karnouskos [Kar04] définissent le paiement mobile comme étant des transactions de paiement électronique effectuées à partir d'un terminal mobile, par exemple un téléphone mobile, un organiseur ou *Personal Digital Assistant* (PDA), une tablette ou un téléphone intelligent *smartphone*. Selon Karnouskos [Kar04], le paiement mobile concerne plus spécifiquement les appareils n'ayant pas uniquement accès au réseau sans fil mais ayant aussi accès au réseau téléphonique. De plus, les produits d'accès qui utilisent le téléphone pour accéder à une procédure d'e-commerce ou paiement par internet existante ne sont pas inclus dans notre définition du paiement mobile.

Pour récapituler, nous considérons que le paiement mobile correspond à tout type de service permettant d'*initier et de confirmer* une transaction pour réaliser un paiement indépendamment d'une procédure d'e-commerce. Les transactions considérées ici correspondent aux paiements marchands, les transferts d'argent entre particuliers et les opérations de gestion de compte comme le retrait et le dépôt.

Selon Karnouskos [Kar04], les différents acteurs impliqués dans le paiement mobile sont les clients, les marchands, les opérateurs téléphoniques, les constructeurs de téléphone, le secteur financier (banque centrale, banques, fournisseurs de carte de crédit,...), le gouvernement en tant que législateur, les fournisseurs de service et les fabricants de logiciel.

Comme le paiement, le paiement mobile peut mettre en œuvre des processus de paiement prépayé, post-payé ou immédiat. Dans le cas des paiement prépayés, nous avons vu que deux types de comptes existent : les comptes prépayés électroniques et les comptes prépayés virtuels. Dans le premier cas, les valeurs sont stockées soit dans le terminal mobile ou la carte SIM. Dans le second cas, les valeurs échangées sont stockées dans un serveur distant qui est accessible via le terminal mobile. Cette dernière catégorie nécessite de pouvoir contacter le serveur distant au moment du paiement.

Du point de vue des technologies utilisées pour transporter les informations des transactions électroniques, les plus courantes sont :

- les SMS (Short Message Service). Ils peuvent être utilisés pour faire des requêtes envers un service, payer via numéro surtaxé ou servir de moyen d'authentification ;
- l'USSD (Unstructured Supplementary Services Delivery). Il s'agit d'une norme permettant notamment de communiquer en temps réel avec un fournisseur de service ;
- le NFC (Near Field Communication). Il s'agit d'une technologie de communication sans fil de courte distance. Elle ne peut être utilisée que dans le cadre de paiement de proximité pour l'échange de données d'authentification ou d'ordre de paiement ;
- les réseaux IP (Internet Protocol). Il s'agit des protocoles de communication sur internet.

En considérant le marché cible et le rôle de certains acteurs, il existe plusieurs modèles de paiement mobile. Le paiement mobile s'adresse à la fois à des populations bancarisées et à des populations non bancarisées. Dans le premier modèle, dit additif, le terminal mobile fournit une forme de paiement supplémentaire en marge de son compte bancaire [LPP08, Por]. Dans ce cas, il soit soit d'une nouvelle manière d'accéder à son service de banque à distance ou d'utiliser sa carte bancaire soit de porte-monnaies électroniques permettant de faciliter certains usages comme les micro-paiements. Dans le deuxième modèle, dit transformationnel, le service de paiement donnent accès à des services dématérialisés de type bancaire à des personnes qui n'ont pas accès à des comptes bancaires [LPP08, Por].

Le rôle plus ou moins dominant de la banque dans la solution de paiement mobile peut aussi être considéré [Kar04, LIS06]. Selon cette approche, deux modèles existent : le modèle bancaire, *bank-centric*, et le modèle opérateur (aussi désigné par Mobile Network Operator, MNO), *MNO-centric*. Dans le modèle *bank-centric*, la banque est responsable du moyen de paiement fourni alors que l'opérateur fournit uniquement le canal qui achemine les informations relatives aux transactions. Dans le modèle *MNO-centric*, l'opérateur gère lui-même le moyen de paiement. Le modèle *MNO-centric* peut être généralisé en considérant que des non-banques qui ne sont pas forcément des opérateurs téléphoniques peuvent fournir le service de paiement. Il existe aussi des modèles composites qui accordent plus ou moins d'importance aux banques ou aux non-banques.

1.1.5 Discussion

Dans cette section, nous avons décrit et défini différentes notions liées à l'écosystème du paiement mobile. L'opération de paiement, la monnaie, les institutions sont détaillées. Le paiement mobile et ses différentes caractéristiques ont également été abordés.

Dans la prochaine partie, nous utilisons les notions concernant l'écosystème du paiement mobile, définies précédemment, pour spécifier les services de paiement mobile sur lesquels se base cette thèse. Dans la suite du manuscrit, nous désignons ce type de service spécifique par l'expression *systèmes* ou *services de transaction sur mobile*.

1.2 Les services de transaction sur mobile actuels à la base de ces travaux

Cette étude concerne les services de paiement sur mobile basés sur un modèle de paiement trois coins et un compte prépayé électronique virtuel. Les comptes sont donc hébergés et gérés sur une plateforme distante qui appartient à un opérateur de téléphonie mobile. Une application dans le terminale mobile permet d'accéder à ce compte et de réaliser diddérentes opérations.

De plus, la monnaie est exprimée sous la forme d'une valeur numérique. Les échanges sont donc des instructions permettant de modifier le solde d'un compte. La monnaie en circulation dans le système est une monnaie électronique privative émise par l'opérateur sous le contrôle d'une banque partenaire et de la banque centrale comme illustré dans la figure 1.4. Nous considérons uniquement le cas où cette banque ne joue pas un rôle prépondéral dans la mise en oeuvre et la gestion du système de transaction sur mobile. Nous sommes donc dans un modèle *MNO-centric*.

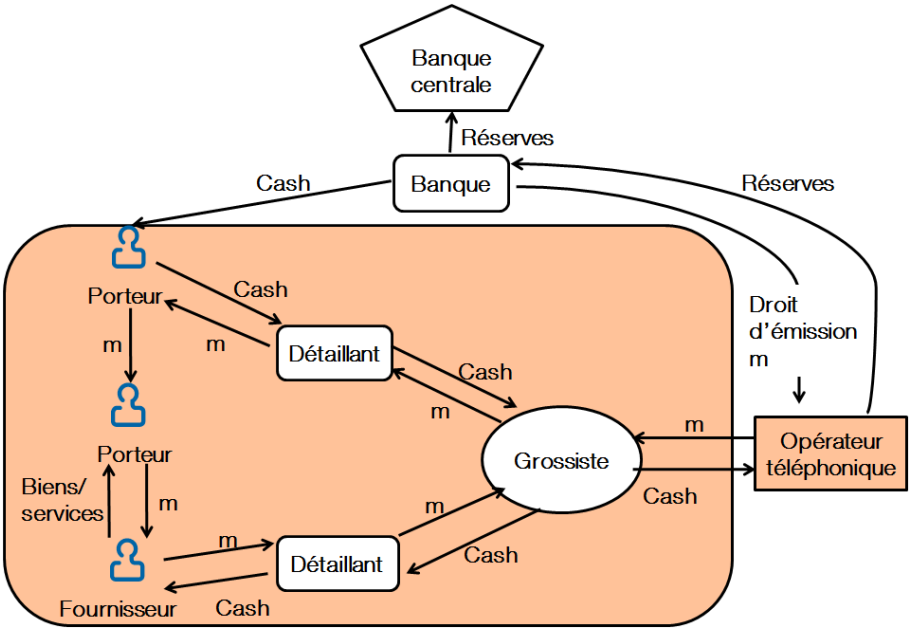


FIGURE 1.4 – Relations entre les acteurs du système de transfert sur mobile, adapté de [JTT10]

Tout type de paiement peut être réalisé dans ce système, qu'il s'agisse de paiement distant ou de proximité, de macro- ou micro-paiement. Dans le cadre de la thèse, nous considérons aussi bien les transactions entre particuliers et entreprise-particuliers.

Dans cette dernière catégorie, nous retenons surtout les achats.

A notre connaissance, dans les services de transaction sur mobile actuels, les mobiles des utilisateurs et la plateforme distante communiquent aujourd'hui à l'aide des technologies USSD et SMS. Cependant, dans cette thèse, nous nous plaçons dans le cadre des réseaux 4G de type tout-IP où le canal transportant l'USSD et les SMS n'existe plus. Le marché cible, bancarisé ou non-bancarisé a peu d'impact sur les travaux de cette thèse.

1.2.1 Organisation et architecture des systèmes actuels

Les services de transaction sur mobiles comprennent plusieurs acteurs dont les relations sont illustrées figure 1.4. La banque centrale et la banque commerciale permettent et contrôlent l'émission de monnaie électronique par l'opérateur téléphonique qui possède le système. En particulier, l'opérateur doit garantir qu'il n'y aura pas de création ou de destruction de monnaie au sein du système. La monnaie électronique est ensuite distribuée à travers un réseau de grossistes à des détaillants ou à des fournisseurs de biens et de services. Ceux-ci la distribuent à leur tour aux porteurs qui réalisent des transferts ou des paiements.

La plupart des acteurs représentés ci-dessus, les porteurs, les fournisseurs de biens ou de services et certains détaillants, accèdent au service grâce à une application sur leur téléphone. Celle-ci communique avec le serveur distant par USSD et SMS en passant par le réseau opérateur. Cette architecture globale est représentée figure 1.5. Cependant, comme mentionné précédemment, dans le cadre de cette thèse, nous nous plaçons dans un contexte où les communications entre le terminal mobile et la plateforme sont réalisés à l'aide de réseaux tout-IP. Certains acteurs autres que le porteur peuvent accéder au service grâce à un ordinateur mais ce cas n'est pas représenté et n'est pas pris en compte dans la thèse.

La plateforme de paiement mobile réunit plusieurs fonctionnalités illustrées figure 1.6. L'utilisateur envoie des requêtes d'opérations au système utilisateur. Celui-ci authentifie l'utilisateur, détermine son profil, ses droits et traite ensuite les différentes requêtes d'opérations. Les demandes de transactions sont ainsi envoyées au système de gestion de compte tandis que d'autres opérations telles que la modification du mot de passe sont traitées au niveau de l'interface. Le système de gestion des comptes gère les autorisations de transactions ainsi que le débit et le crédit des différents comptes.

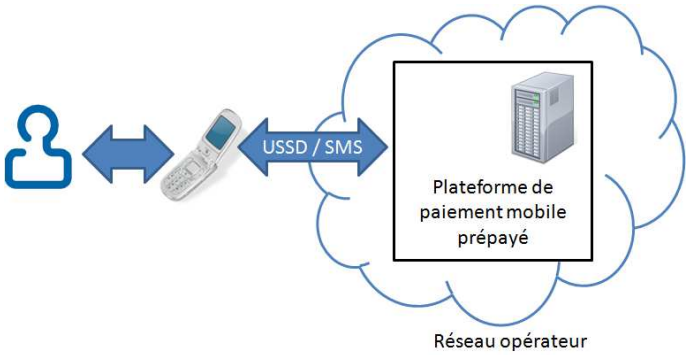


FIGURE 1.5 – Architecture globale

Les succès ou échecs des différentes opérations et les transactions sont enregistrés dans des entrepôts de données et notifiés à l'utilisateur.

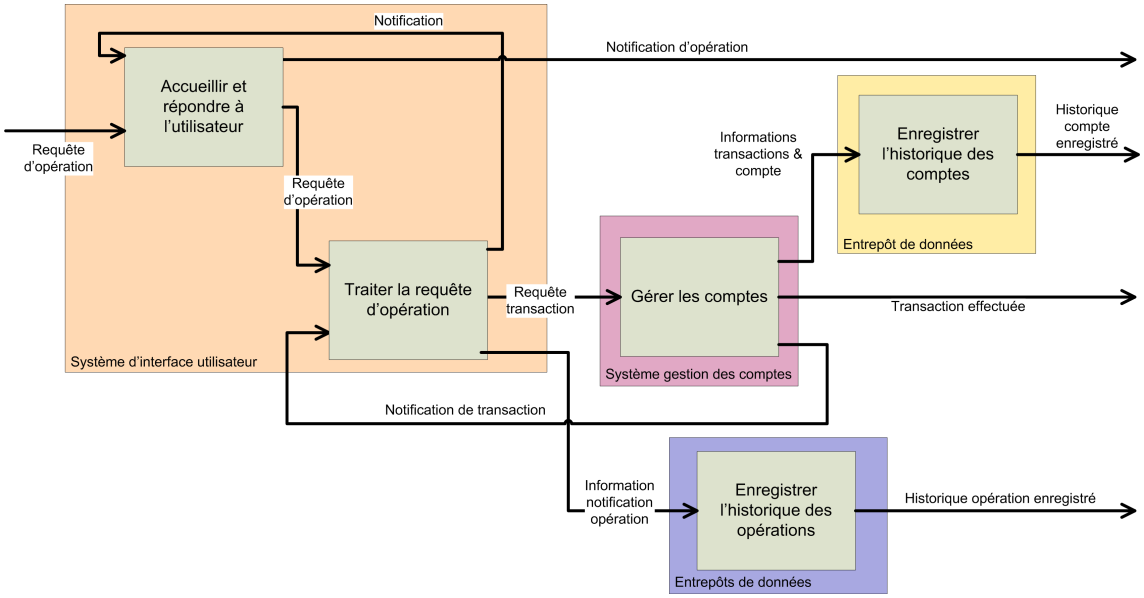


FIGURE 1.6 – Architecture fonctionnelle

1.2.2 La sécurité des services de transaction sur mobiles actuels

Les services de transaction sur mobiles actuels utilisent l'USSD et les SMS qui passent par les canaux de signalisation du réseau fourni par l'opérateur. Cela induit un certain nombre de limitations qui ont un impact sur la sécurité et l'utilisation des services.

Tout d'abord, l'USSD, dispose d'un débit de 800 bits par seconde. La transmission de données de forte taille ne peut être envisagée par ce bais à cause de sa lenteur. Les

menus chargés par l'USSD sont donc très simplistes afin d'éviter un temps d'attente trop long et inacceptable dans le cadre du paiement. En conséquence, ces menus sont peu ergonomiques. Cela pourrait être amélioré si les messages transitaient par l'ADSL qui a un débit de minimum de 1 Mbits par seconde ou la 3G/4G avec un débit théorique de 380 kbits à 100 Mbits par seconde.

De plus, les réseaux 2G sont vulnérables aux attaques *man-in the middle* entre une antenne-relais GSM (BTS, Base Tranceiver Station) et un mobile. Dans le cas de la 2G, il n'y a pas d'authentification mutuelle [NP09]. Comme c'est l'antenne-relais qui choisit les protocoles de sécurité utilisés, elle peut imposer au mobile des protocoles de sécurité faible. Cette attaque paraît d'autant plus probable aujourd'hui que le matériel pour la réaliser est devenu abordable ou accessible de manière libre [NP09]. Cette attaque a été corrigée dans le cadre des réseaux de troisième (3G) et quatrième génération (4G). Cependant, elle est toujours possible puisque la sécurité de la 3G peut être dégradée afin d'assurer la compatibilité avec des réseau ou des terminaux 2G.

Aujourd'hui, les données ne sont pas sécurisées de bout-en-bout entre le serveur et la carte SIM. Les données transmises par un mobile sont chiffrées uniquement entre le mobile et la première antenne rencontrée. Ensuite, ces données transitent en clair puisqu'elles sont censées être dans le réseau qui appartient à l'opérateur et qui ne lui appartient pas. Ce cas est illustré figure 1.7. Il existe cependant des cas où l'opérateur fait appel à des réseaux qu'il ne maîtrise pas comme le montre figure 1.8. C'est le cas pour l'itinérance par exemple. Il existe également des situations où l'opérateur choisit de faire passer ses données d'une antenne réseau vers un réseau filaire tout-IP qu'il ne maîtrise pas pour ensuite les faire revenir dans un réseau qu'il contrôle.

Finalement, nous nous plaçons dans le contexte où le canal USSD et SMS serait supprimé lors du passage des réseaux 3G aux réseaux 4G en tout-IP. D'après l'organisation de standardisation 3GPP, Third Generation Partnership Project, les SMS (et donc l'USSD) seront encapsulés dans des paquets IP [3GP09]. Il sera possible d'adapter les services actuels de transaction sur terminaux mobiles aux réseaux tout-IP en utilisant cette solution sans modifier les procédures et l'architecture actuelles. Cependant, cette solution ne tire pas profit des nouvelles capacités des réseaux tout-IP et ne fournit pas une sécurité de bout-en-bout au niveau applicatif.

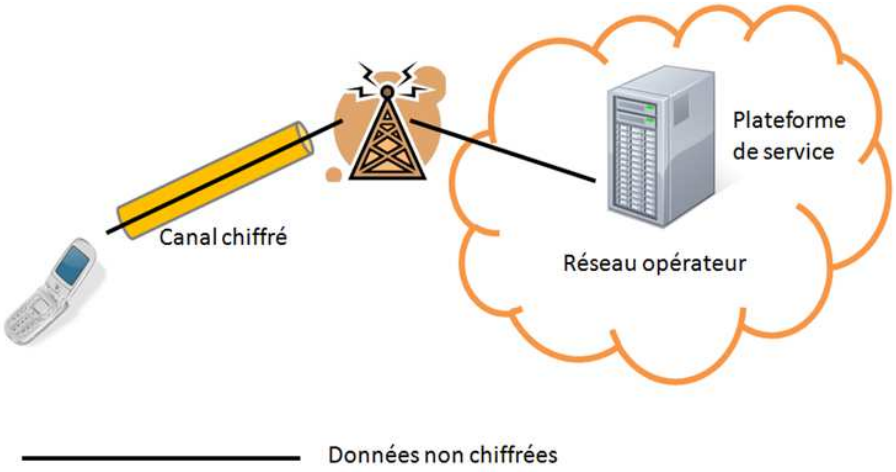


FIGURE 1.7 – Les données ne passent que par le réseau de l'opérateur à qui appartient la plateforme de service.

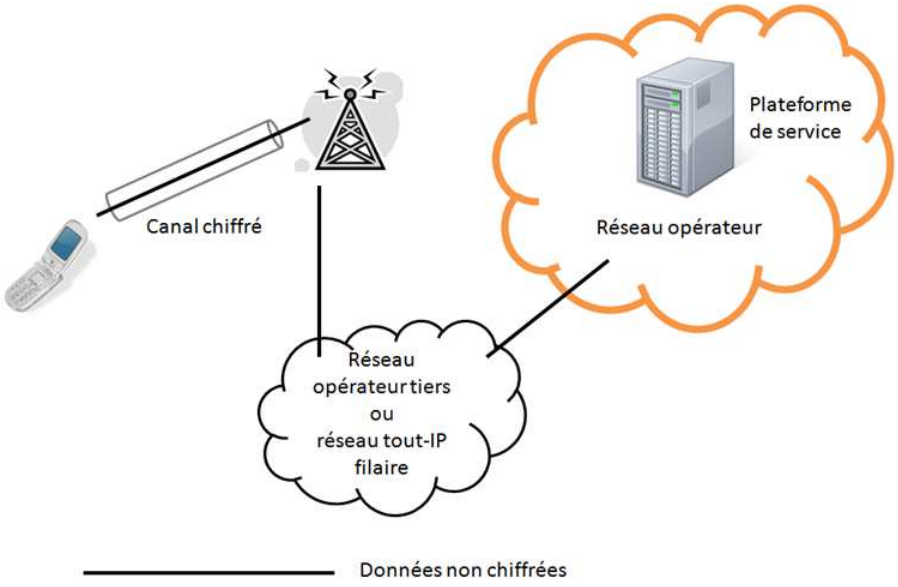


FIGURE 1.8 – Les données passent par un autre réseau que celui de l'opérateur à qui appartient la plateforme de service

1.2.3 Discussion

Les définitions présentées en section 1.1 ont été utilisées pour caractériser les systèmes étudiés dans cette thèse. La section 1.2.2 montre que l'utilisation des canaux USSD et SMS dans les services de transaction sur mobile actuels présentent plusieurs limitations. De plus, elles ne permettent pas de bâtir des services dont la sécurité est assurée de bout-en-bout.

Ces différentes vulnérabilités peuvent être exploitées afin de réaliser des fraudes. Dans la prochaine section, nous définissions ce qu'est la fraude et à quoi correspond la gestion de la fraude.

1.3 La fraude et la gestion de la fraude

1.3.1 La fraude

Le but de cette partie est de définir la fraude. Pour cela, la fraude sera tout d'abord positionnée parmi les risques concernant les systèmes de paiement. Ensuite, la définition de la fraude sur laquelle se basera la thèse sera présentée.

La fraude dans les risques liés aux systèmes de paiement

La Banque Centrale Européenne définit plusieurs types de risques pouvant toucher le paiement [Ban10] :

- Le risque de crédit. C'est le risque que le débiteur ne paie pas sa dette à l'échéance voulue ;
- Le risque de liquidité. Cela correspond au risque de ne pas avoir suffisamment de fonds pour faire face à ses engagements à leur échéance. Par exemple, si les clients d'une banque retirent tous leurs dépôts en même temps, la banque peut ne pas avoir suffisamment de liquidité pour faire face à cette demande. Un tel scénario pourrait entraîner la banqueroute de la banque ;
- Le risque opérationnel. Il correspond au risque de subir des pertes suite à un dysfonctionnement dans des processus internes, dans le système, les composants techniques ou dans des facteurs externes. Le risque opérationnel ne prend pas uniquement en compte des défaillances techniques mais aussi des erreurs d'employés, des fraudes ou l'indisponibilité d'acteurs externes ;
- Le risque légal. Ce risque est lié à l'application et l'interprétation des lois. Il peut par exemple s'agir d'une loi mal appliquée ou mal anticipée ;
- Le risque systémique. Il s'agit du risque que l'ensemble du système soit affecté par un problème touchant l'un des acteurs du système.

La fraude fait donc partie de la catégorie de risque opérationnel. Selon Aglietta et Scialcom [AS02], cette catégorie correspond aux pertes potentielles liées à des déficiences en matière de fiabilité et d'intégrité des systèmes électroniques. Selon l'ouvrage de la banque centrale européenne sur les systèmes de paiements [Ban10], une définition étendue du risque opérationnel est fournie par le Comité de Bâles

sur la supervision bancaire dans le cadre de Bâles 2. Ici, le risque opérationnel est considéré comme étant *le risque de pertes provenant de processus internes inadéquats ou défaillants, de personnes et systèmes ou d'événements externes*. Cette définition signifie que le risque opérationnel correspond à l'ensemble des vulnérabilités pouvant mettre en péril les ressources financières. Cela recouvre des domaines très larges tels que les erreurs humaines, les défaillances des systèmes, les problèmes liés à la gestion du personnel, les accidents, les litiges commerciaux ainsi que les fraudes et les malveillances.

Maintenant que le contexte dans lequel la fraude s'inscrit a été abordé, celle-ci va être définie.

Caractérisation de la fraude

La fraude est un terme qui recouvre plusieurs réalités selon le domaine où il est abordé. Dans le cas des télécoms, la fraude représente l'utilisation du service sans contrepartie. Dans le domaine de l'informatique, la fraude prend place dans les ordinateurs ou sur le réseau. Elle correspond à l'utilisation de moyens informatiques pour mettre en place des escroqueries, le piratage de systèmes informatiques, le vol de données ou encore la violation des droits d'auteur ou de *copyright*. Dans le domaine financier, la fraude prend différentes formes. Par exemple, la manipulation d'investisseurs dans le but d'obtenir un gain constitue une fraude ainsi que la contrefaçon de monnaie, l'omission de certaines données afin d'obtenir un prêt.

Comme la fraude admet une pluralité de définitions, il convient de définir ce qui sera considéré comme étant une fraude dans le cadre de la thèse. Pour cela, différents aspects de la fraude des domaines ci-dessus seront repris et assemblés. La notion de risque sera aussi examinée afin de caractériser la fraude.

Condamin *et coll.* [CLP06] définissent que le risque est une menace visant une certaine cible et ayant un certain impact potentiel. Il existe des menaces de différentes natures. Celles-ci sont représentées dans la figure 1.9, adaptée de l'ouvrage [Lou05]. La menace de type opérationnel correspond à une défaillance technique ou organisationnelle dans un système. La menace de type naturel correspond à des événements naturels comme la pluie ou un séisme par exemple. La menace de type économique correspond à des événements économiques tels qu'une hausse de prix, l'augmentation d'impôts. Enfin, la menace d'origine humaine correspond à des actions réalisées par des personnes. La description de cette menace est raffinée pour prendre en compte la

volonté de nuire et le but de l'action.

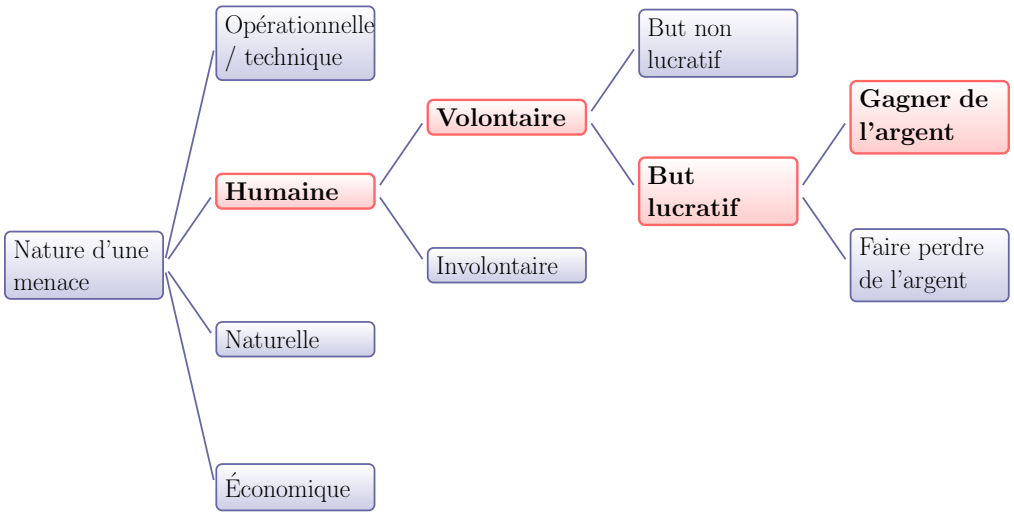


FIGURE 1.9 – Nature des menaces, adaptée de [Lou05]

Dans le cadre de ces travaux, nous étudions les risques qui ont pour cible les ressources financières du système de paiement mobile et qui sont associés à une menace humaine volontaire avec un but lucratif.

A partir de la définition du risque et des menaces de Condamin *et coll.* [CLP06] ainsi que des différentes définitions de la fraude vues précédemment, nous avons défini que la fraude considérée dans le cadre de la thèse est : *une action qui est menée volontairement par une personne ou un ensemble de personnes externes ou internes au système et qui utilise une vulnérabilité du système pour obtenir un gain.* Cela correspond au chemin rouge dans la taxonomie des menaces, décrite dans la figure 1.9. Cette définition nous servira de base pour spécifier les risques qui sont spécifiques aux services de transaction sur mobiles et que nous allons considérer dans le cadre de ce manuscrit.

Après avoir défini ce qu'est la fraude, nous détaillons maintenant la gestion de la fraude.

1.3.2 La gestion de la fraude

La gestion de la fraude s'inscrit dans une démarche de gestion des risques. De nombreuses méthodes et outils permettent de gérer tous types de risques. La base

commune à ces méthodes est la mise en place d'un système d'amélioration continue. Cela correspond à un processus cyclique d'identification, d'évaluation et de traitement des risques. L'objectif de cette démarche est d'identifier les risques inacceptables et de mener des actions pour que ces risques deviennent acceptables.

La figure 1.10 représente notre définition du cycle de la gestion de la fraude dans un système. En haut à gauche du schéma, on part d'un état où le système est supposé être le plus sûr possible. Ce système dispose d'un certain nombre de mesures de sécurité permettant de bloquer des fraudes. Cependant, plusieurs fraudes peuvent tirer profit de vulnérabilités résiduelles pour passer au travers de ces mesures de sécurité. L'identification de ces vulnérabilités peut se faire à travers plusieurs moyens. Des audits peuvent être menés sur le système, ou les événements enregistrés peuvent être analysés afin de détecter la présence de fraudes connues ou d'éléments anormaux. Il est également possible d'avoir un retour d'expérience extérieur. Celui-ci peut concerner des systèmes similaires ou provenir d'acteurs en amont ou en aval du système. Le retour d'expérience extérieur peut également résulter d'études concernant les technologies utilisées dans le système.

À partir du moment où des vulnérabilités et donc des risques sont identifiés, une analyse et une évaluation du risque sont réalisées. Suite à cela, les gestionnaires du système peuvent décider d'accepter, d'assurer ou de traiter le risque. Ce dernier cas implique une modification du système et des mesures de protection du risque. À la suite d'un cycle, le système revient à l'état sûr.

Dans le cycle de gestion de la fraude, les contributions de cette thèse se situent au niveau des moyens de sécurisation et des méthodes d'analyse des événements. Une architecture et des protocoles permettant de réaliser des transactions sécurisées de bout-en-bout sont proposés et une étude des algorithmes de classification est réalisée.

1.4 Discussion

Dans ce chapitre, le contexte de la thèse a été défini. Pour cela nous avons décrit diverses notions liées au paiement, à la monnaie et au paiement mobile. À l'aide de celles-ci, nous avons spécifié les systèmes auxquels les travaux de cette thèse s'appliquent. Ceux-ci sont basés sur l'utilisation de terminaux mobiles, de monnaie électronique privative et de comptes virtuels prépayés. Le modèle de paiement considéré est le modèle trois-coins.

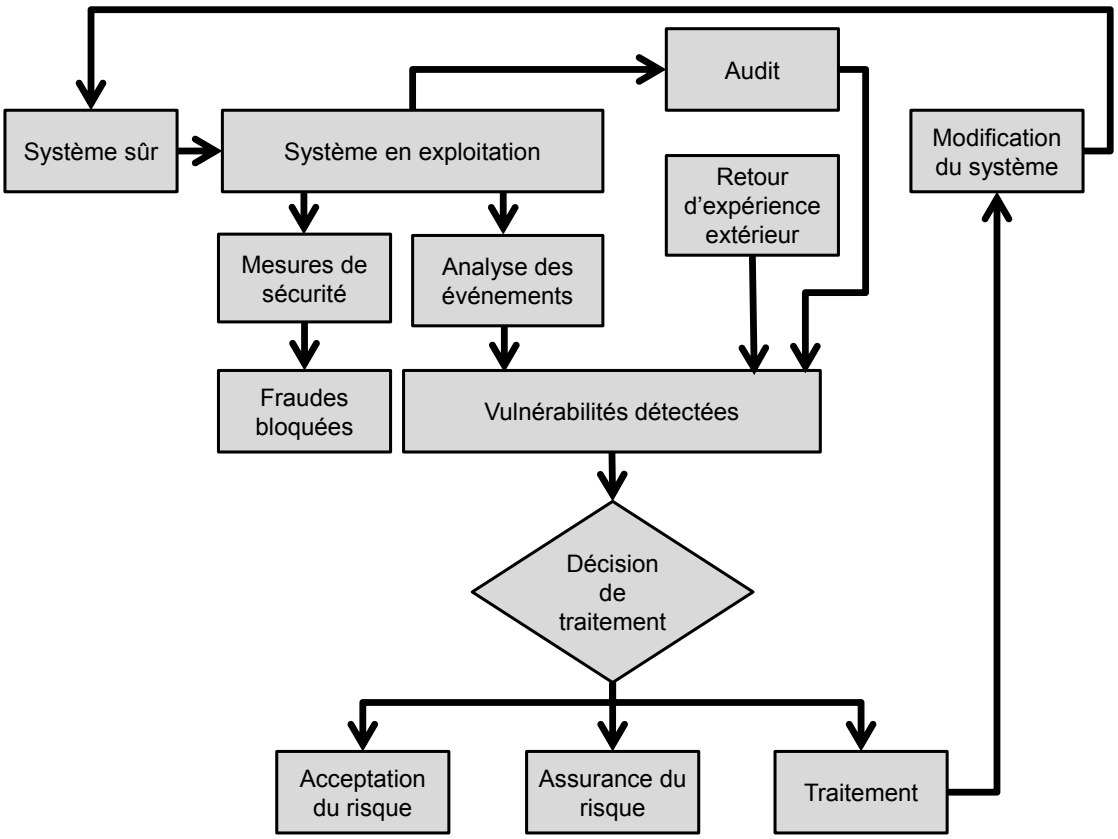


FIGURE 1.10 – Cycle de la gestion de la fraude d'un système

Dans la section 1.2.2, nous avons exposé différents arguments qui montrent que l'architecture actuelle des systèmes ne permet pas d'assurer la sécurité de bout-en-bout de ces services de paiement. Les caractéristiques des réseaux tout-IP peuvent être utilisées pour bâtir des services dont la sécurité est assurée de bout-en-bout entre une plateforme de service et l'application dans la carte SIM.

Ces limitations peuvent être des vulnérabilités du système étudié et peuvent donc être exploitées afin de réaliser des fraudes.

Parmi les différents éléments du cycle de gestion de la fraude, les contributions de la thèse se concentrent sur deux axes de recherche. Le premier concerne les architectures sécurisées pour les systèmes de transactions sur terminaux mobiles. Le deuxième correspond à l'adaptation de méthodes de classification au problème de détection de la fraude dans les systèmes de transactions sur terminaux mobiles.

Dans le cadre cette thèse, nous nous plaçons dans un contexte oùles réseaux utilisés sont des réseaux de quatrième génération de type tout-IP. Contrairement au premier axe de l'étude, le second axe est indépendant de ce contexte. Les travaux concernant la détection sont donc applicables aux systèmes de transaction sur mobile actuels.

Le chapitre 2 est un état de l'art concernant les architectures sécurisées de paiement mobile et les algorithmes de classification. Comme nous le montrerons, ces algorithmes permettent de répondre au problème de détection de fraude.

Chapitre 2

Etat de l'art sur la sécurisation des systèmes de transactions sur terminaux mobiles

L'objectif de ce chapitre est de dresser un panorama des approches et solutions existantes pour sécuriser les services de transactions sur mobiles. Tout d'abord, différentes catégories de risques de fraudes sont identifiées. Ensuite, un état de l'art des architectures de sécurité pour le paiement mobile est réalisé ainsi qu'un état de l'art concernant les algorithmes de classification.

Sommaire

2.1 Les risques de fraudes dans les services de transaction sur mobile

Les services de transaction sur mobile sont encore récents et il existe peu de travaux traitant de la fraude dans ce domaine. La littérature dans le domaine de la détection traite surtout de la fraude dans le domaine bancaire et dans le modèle quatre-coins qui ne correspond pas à notre contexte. Nous présentons ici les risques

que nous avons identifiés pour les services de transactions sur mobiles à partir de l'étude de la fraude dans le domaine bancaire, des télécommunications et des paiements mobiles prépayés actuels. Nous nous sommes restreints aux risques ayant des conséquences sur les porteurs ou les impliquant. Les conséquences pour l'opérateur de téléphonie mobile sont également considérées tout en restant dans le cadre que nous avons défini dans le chapitre 1. Nous ne prenons donc par exemple pas en compte les attaques, telles que le déni de service, qui ciblent le système de paiement de l'opérateur pour lui nuire sans forcément en tirer de gains.

Plusieurs cas de *tromperies* ou d'escroqueries ont été observés dans le domaine du paiement [K.J10] ou dans les systèmes de transactions sur mobile actuels [Mud13]. Ces fraudes consistent à convaincre, avec des arguments fallacieux, un porteur de transférer de l'argent ou un marchand qu'un paiement a bien été réalisé.

Il est également possible qu'un attaquant s'introduise dans le système d'information et réalise des opérations illégales. Cette *compromission du système* aussi appelée *server-side attacks* est définie pour l'ensemble du domaine des paiements dans [K.J10]. L'attaquant peut effectuer des opérations sur les comptes des porteurs, exécutant ainsi une forme de fraude comportementale. Il peut également créer ou supprimer de la monnaie électronique ou encore prélever une commission sur chaque transaction. Cette forme d'attaque peut également être réalisée par un acteur interne du service qui utilise ou outrepasse ses droits pour réaliser des opérations malveillantes. Il s'agit là de la *fraude interne* [K.J10].

Il existe plusieurs formes d'*usurpation d'identité*. La première, la fraude à la souscription, consiste à souscrire au service sous une fausse identité. Le but est de cacher des activités illégales par exemple. La seconde consiste à utiliser des vulnérabilités techniques ou organisationnelles pour utiliser un compte à l'insu de son propriétaire. Ceci pourrait être réalisé en installant un programme malveillant sur le terminal cible par exemple. Dans ce cas, le comportement du fraudeur vient se mêler à celui du porteur légitime. Cette fraude existe dans le domaine des télécommunications sous le nom de fraude superposée ou *superimposed fraud* [FP97]. Elle existe aussi dans le domaine bancaire sous le nom de fraude comportementale ou *behavioral fraud* en anglais [BJTW11, DAP09]. Nous retiendrons le terme fraude comportementale dans la suite de ce document.

L'analyse de risques réalisée par Microsave [Mud13] comprend plusieurs risques

liés à de la fraude comportementale, cependant l'utilisation de vulnérabilités du réseau ou du téléphone n'est pas mentionnée. La raison est que cette étude vise les solutions de transfert sur mobiles existantes qui ne sont basées ni sur des réseaux tout-IP, ni sur une large utilisation de smartphones. le contexte où se place la thèse se rapproche de celui des paiements distants sur internet où ni le réseau ni le terminal utilisé pour réaliser la transaction ne sont de confiance. Dans ce contexte, la fraude comportementale est un risque important et c'est pour cela que nous considérons ce type de fraude pour la suite.

Après avoir déterminé les types de risques que nous prenons en compte, nous décrivons maintenant les architectures de sécurité existantes dans le domaine du paiement mobile.

2.2 Architectures de sécurité de paiement mobile

L'objectif de cette section est d'étudier les différents moyens mis en œuvre pour sécuriser les paiements mobiles aujourd'hui. Pour cela, nous détaillons tout d'abord les différents composants des terminaux mobiles qui fournissent des services de sécurité. Ensuite, nous décrivons différentes architectures dans lesquelles les terminaux mobiles s'intègrent et qui assurent la sécurité des interactions entre les différents acteurs d'un paiement.

Nous supposons ici que les acteurs en jeu dans le paiement mobile sont les suivants :

- le payeur, l'expéditeur, qui possède un terminal mobile, souscrit à un service de paiement et réalise un paiement ;
- le payé ou le destinataire, qui reçoit un paiement. ;
- la plate-forme de paiement reçoit une instruction de paiement. Elle est responsable de la réalisation de cette opération. Son rôle est différent dans le cas d'un modèle quatre-coins ou trois-coins, définis dans le chapitre 1 mais nous faisons abstraction de ces différences dans cette partie.

Nous ne prenons pas en considération des applications telles que Paypal Mobile [1] ou Buyster [2] qui enregistrent les informations des cartes bancaires des clients pour

1. https://www.paypal.com/fr/webapps/mpp/mobile
2. http://www.buyster.fr/?xtor=17

débiter les comptes qui leur sont associés en cas de paiement. Le service de Paypal qui gère de la monnaie électronique prépayée, comme c'est le cas de notre contexte, n'est pas non plus pris en compte car il s'agit d'une extension du service en ligne et ne s'appuie pas sur des fonctionnalités spécifiques au mobile. Finalement, nous ne prenons pas non plus en compte les cas où les paiements sont réalisés à travers des messages SMS surtaxés ou à travers la facture téléphonique.

Ces limitations sont principalement dues à notre définition du paiement sur terminaux mobiles dans le chapitre 1. Selon celle-ci, une transaction correspond à un paiement sur terminaux mobiles si celle-ci est initiée et confirmée à l'aide du mobile et si elle ne correspond pas à un processus existant déjà sur internet pour un usage avec un ordinateur.

2.2.1 Sécurité des terminaux mobiles

Trois composants mobiles peuvent être utilisés pour fournir des services de sécurité dans les terminaux mobiles. Il s'agit des systèmes d'exploitation des terminaux mobiles, de l'environnement d'exécution sécurisée ou *TEE* pour *Trusted Execution Environment* en anglais et des éléments matériels sécurisés ou *SE* pour *Secure Element* en anglais.

Les systèmes d'exploitation, Operating System ou *OS* en anglais, sont des composants logiciels qui permettent à des applications d'utiliser les ressources matérielles, comme l'écran, le clavier, la mémoire ou l'unité centrale de traitement des terminaux mobiles. Comme le montrent Gaber *et coll.* [GP10], les principaux systèmes d'exploitation des terminaux mobiles, Android, Symbian, iOS, Windows Mobile, fournissent un certain nombre de services de sécurité en terme de gestion de la mémoire, gestion des permissions et de cryptographie. Cependant, ces services sont implémentés différemment selon les plateformes et restent insuffisants. En effet, ces systèmes d'exploitation fournissent des services riches et comportent en conséquence des milliers de lignes de code. Cette grande taille ne permet pas de garantir l'absence de faille et rend impossible une certification complète. De plus, l'ouverture de ces plateformes au téléchargement d'applications non certifiées rend la diffusion de programmes malveillants très facile. Même les systèmes d'exploitation fermés comme iOS ne sont pas épargnés puisqu'il est tout de même possible d'en détourner les mesures de sécurité et d'y installer des programmes malveillants [WLL+13]. La sécurité fournie par les systèmes d'exploitation n'est pas suffisante pour garantir la

sécurité d'applications sensibles comme des applications de paiement mobile. Ces difficultés peuvent être contournées grâce à l'environnement d'exécution sécurisée et l'élément de sécurité que nous proposons d'utiliser conjointement.

L'environnement d'exécution sécurisée est un élément matériel et logiciel présent sur les terminaux mobiles. Il fournit un environnement d'exécution isolé du système d'exploitation traditionnel. Contrairement aux systèmes d'exploitation, l'intégrité, l'authenticité et la confidentialité des applications qui s'exécutent dans l'environnement d'exécution sécurisée et leurs données sont protégées [Glo11b]. Ainsi, cet environnement gère les périphériques de manière sûre et n'est pas limité en capacité comme les éléments sécurisés. Cependant, l'environnement d'exécution sécurisée n'est pas sécurisée, contrairement à l'élément sécurisé [Glo]. La spécification Global Platform [Glo11b] décrit l'architecture des environnements d'exécution sécurisée. TrustZone® est un exemple d'un tel composant développé par la société ARM. Cette société s'est associée en 2012 à Gemalto et Giesecke & Devrient pour former la société Trustonic® et développer cette technologie. Cela implique que celle-ci devrait être largement développée et distribuée dans le futur. Cette évolution a déjà commencé puisque les processeurs TrustZone sont utilisés dans les processeurs Exynos de Samsung [3] qui équippent les Galaxy Note II et les Galaxy S3 [4].

L'élément sécurisé est un composant matériel difficilement attaquable. Ses fonctionnalités garantissent un fort niveau de sécurité. Trois types d'éléments sécurisés existent. Les cartes SIM sont les plus répandues et sont disponibles dans tous les téléphones. Il existe aussi des éléments sécurisés embarqués, *embedded SE*, comme les *TPM*, Trusted Platform Module, pour les ordinateurs [TCG11]. Ceux-ci sont soudés dans le terminal mobile, ce qui implique un coût supplémentaire à la carte SIM. Le dernier type de SE est la carte SD sécurisée *Secure SD Card*. La limitation actuelle des éléments sécurisés est qu'ils ne gèrent pas les périphériques. Une exception correspond à la puce NFC qui intègre un élément sécurisé qui peut donc communiquer en NFC de manière sécurisée. Il n'est donc pas possible de garantir l'intégrité des informations envoyées et reçues via les interfaces homme-machine. De même, l'élément sécurisé doit s'appuyer sur les services offerts par le mobile et le réseau pour dialoguer avec une plateforme distance.

Combiner l'utilisation des éléments sécurisés et des environnements d'exécution sécurisés permet d'obtenir à la fois 1) une zone de stockage et de calcul inviolable

3. http ://www.trustonic.com/partners/samsung
4. http ://www.samsung.com/global/business/semiconductor/minisite/Exynos/products4quad.html

pour les opérations et données très sensibles, 2) une zone de stockage et de calcul de plus grande capacité pour des calculs ou des données moins sensibles, 3) une gestion sécurisée des périphériques qui n'est pas possible avec les systèmes d'exploitation de terminaux mobiles actuels. Dans l'architecture que nous proposons, l'élément sécurisé et l'environnement d'exécution sécurisée dialoguent ensemble, ce qui permet d'effectuer de manière sécurisée les calculs et les communications avec l'utilisateur.

2.2.2 Interaction des terminaux avec les plateformes de paiement

Les architectures de paiement sur mobile se divisent en trois catégories selon l'interaction qui existe entre les différents acteurs du paiement [CKST01]. Si, comme c'est le cas figure 2.1, les trois acteurs sont connectés et interagissent lors de l'opération de paiement, l'architecture est dite tout-connectée. Si, comme en figure 2.2, le payé et le payeur interagissent et seul l'un d'entre eux est connecté à la plateforme de paiement, il s'agit d'une architecture semi-connectée. Il existe deux catégories d'architectures semi-connectées [CKST01]. Celle où le porteur est l'entité connectée à la plateforme de paiement est appelée *user-centric*. Celle où le marchand est l'entité connectée à la plateforme de paiement est appelée *kiosk-centric*. Si le payeur et le payé interagissent mais qu'aucun d'entre eux n'est connecté à la plateforme de paiement, l'architecture est dite déconnectée (bien qu'une connexion existe entre le payeur et le payé). Elle est représentée en figure 2.3.

Cette partie a pour but d'étudier les différentes architectures qui viennent d'être décrites. Pour cela, nous replaçons différents services de paiement mobile qui existent aujourd'hui dans ces trois catégories et nous étudions des exemples de chacune de ces architectures. Finalement, nous discutons du positionnement de notre problème par rapport aux architectures existantes.

La plupart des systèmes de transaction sur terminaux mobiles déployés actuellement comme Orange Money ou M-Pesa sont basés sur une architecture tout-connectée. Généralement, le marchand initie la transaction en envoyant au serveur le montant de la transaction et le numéro de téléphone de son client entre autres. Le client est ensuite contacté par la plateforme de paiement afin qu'il confirme le paiement. Nous considérons que le cas du transfert entre particuliers est un cas spécifique d'utilisation de cette architecture tout-conectée. Les porteurs et les marchands correspondent avec la plateforme de paiement à l'aide de SMS ou d'USSD. La sécurité de ces systèmes est basée sur les fonctionnalités du réseau mais comme nous l'avons vu dans le chapitre

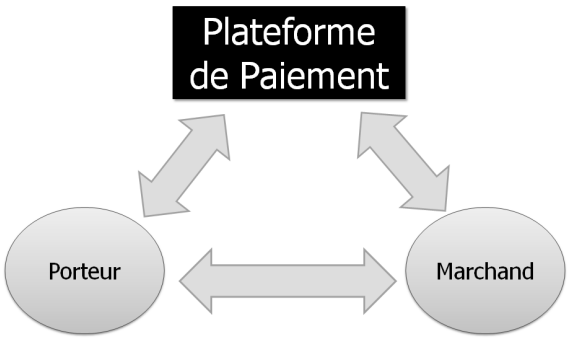
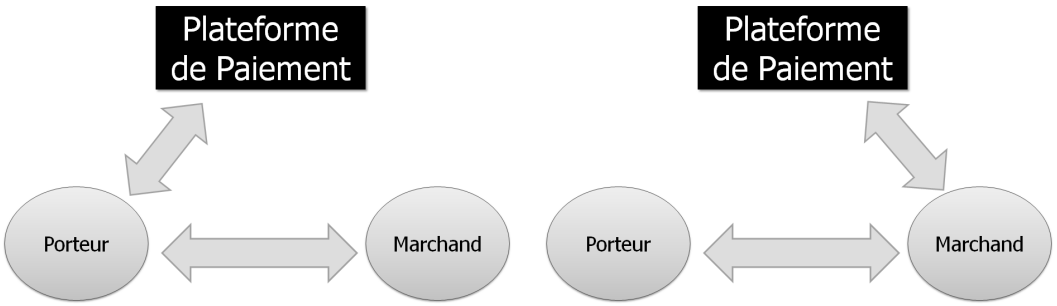


FIGURE 2.1 – Architecture tout-connectée, adaptée de [CKST01]



(a) Connexion via le porteur - *user-centric* (b) Connexion via le marchand - *kiosk-centric*

FIGURE 2.2 – Architecture semi-connectée, adaptée de [CKST01]

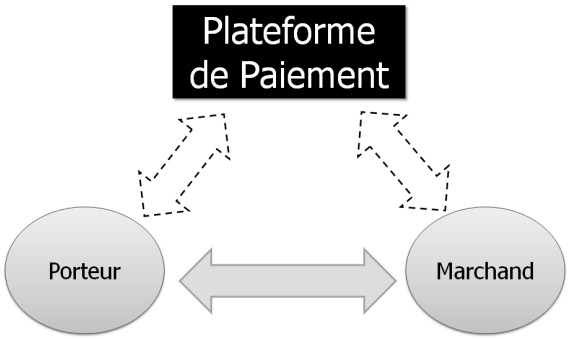


FIGURE 2.3 – Architecture tout-déconnectée adaptée de [CKST01]

1, cela représente des limitations et des faiblesses.

Parmi les architectures tout-connectées pouvant être déployées sur des réseaux tout-IP, certaines, [GKR+09, HHH06, KHVC04, MY08, ZFM08], sont basées sur une infrastructure à clé publique, *PKI*. D'autres, [FBLR08, SS12], utilisent des clés secrètes pour sécuriser les communications. Seules les architectures présentées dans [HHH06, KHVC04] utilisent un SE. Ces deux architectures ne peuvent pas s'appli-

quer à notre contexte. Celle de Karnouskos *et coll.* [KHVC04] inclut les banques qui ne font pas partie de notre contexte. Quant à Hassinen *et coll.* [HHH06], leur architecture utilise les comptes bancaires des clients pour transmettre leur identité au marchand ce qui n'est pas le cas dans notre contexte.

Les systèmes de paiement mobile qui ont une architecture semi-connectée correspondent majoritairement aux cas où le mobile se substitue à la carte bancaire. Il s'agit par exemple d'une partie de l'application Google Wallet [5] ou de Cityzi [6] en France. Dans ce cas, un élément sécurisé dans le mobile contient une application similaire à celle présente sur la carte bancaire. Ce SE dialogue ensuite avec un Terminal de Paiement Electronique, *TPE* chez le marchand. Le terminal de paiement électronique dialogue avec la plateforme de paiement si nécessaire. La transaction suit alors le même parcours qu'une transaction réalisée par carte bancaire et la sécurité de la communication est assurée par les fonctionnalités du terminal de paiement électronique. Cette approche n'est pas adaptée à notre contexte car nous prenons comme hypothèse que le marchand utilise lui aussi un mobile et non un terminal de paiement électronique. De plus, le système que nous considérons est un système 3-coins géré par un opérateur téléphonique. Toute l'infrastructure bancaire qui assure la sécurité des instructions de paiement après le TPE n'est donc pas la même dans notre cas.

Les architectures [CHM+10, KLK09, NSCov] proposent des architectures *kiosk-centric* et [IC07] propose une architecture qui peut être centrée sur le porteur ou sur le marchand. Parmi ces architectures, [CHM+10, IC07] ne spécifient pas où sont stockées les clés et les informations sensibles nécessaires pour assurer la sécurité des paiements. Nguyen *et coll.* [NSCov] utilisent le mobile pour réaliser cette tâche, ce qui est une hypothèse inacceptable aujourd'hui. Seule l'architecture proposée par [KLK09] se base sur l'utilisation d'un élément sécurisé. Cette architecture n'est pas adaptée puisqu'elle se base sur les infrastructures bancaires existantes pour gérer la transaction après traitement chez le marchand. L'architecture proposée dans [CHM+10] n'est pas adaptée à notre cas d'usage puisqu'elle s'appuie sur des fonctionnalités des réseaux GSM et 3G dont nous cherchons à être indépendants en terme de sécurité. L'approche des architectures proposées par Isaac et Camara [IC07] ne sont également pas adaptées à notre contexte puisqu'elle utilise des informations bancaires. A notre connaissance, aucun système de transaction sur mobiles ne se base sur une architecture semi-connectée aujourd'hui.

5. http://www.google.com/wallet/
6. http://www.cityzi.fr/

Les architectures tout-déconnectées correspondent surtout aux applications de porte-monnaie électronique. Il s'agit par exemple des applications Edy et Nanaco [BH07] comprises dans les services fournis par *Osaifu Keitai* au Japon [Doc]. *Osaifu Keitai* est un ensemble de services allant du paiement mobile, à la gestion de cartes de fidélité ou d'identités. La sécurité de ces services est basée sur la technologie Felica qui est une implémentation particulière de NFC. La monnaie électronique est stockée dans le SE Felica et la sécurité de l'échange est assurée par le NFC et le SE.

Ce type d'applications existe déjà sur des cartes à puce, il s'agit par exemple des systèmes Moneo [mon] ou Geldkarte [gel]. Cependant, ces systèmes permettent uniquement le paiement à un marchand en utilisant un terminal NFC spécifique. Le paiement entre porteurs n'est pas possible. Van Damme *et coll.* [VDWKP09] proposent une architecture de paiement tout-déconnectée qui permet le paiement marchand et le transfert entre particuliers. Il est basé sur l'utilisation de coupons pour transporter la valeur monétaire [VDWKP09]. Ce type d'approche n'est pas compatible avec notre cas d'usage parce que nous souhaitons lier le service de paiement tout-déconnecté avec le service de paiement tout- ou semi-connecté qui n'est pas basé sur l'utilisation de coupons.

A notre connaissance, aucun système de transaction sur mobile ne se base sur une architecture tout-déconnectée aujourd'hui.

2.2.3 Discussion sur les architectures de sécurité

Les systèmes de transaction sur mobiles déployés aujourd'hui et proches de notre contexte d'étude ont une architecture tout-connectée. Nous souhaitons cependant faire évoluer ces processus pour assurer l'anonymat du client et pour assurer une sécurité de bout-en-bout entre l'application de paiement et la plateforme. Nous pouvons également tirer profit des réseaux tout-IP pour fournir de nouvelles fonctionnalités et de nouveaux services. Dans le cas de l'architecture toutconnectée, l'utilisation des réseaux tout-IP impose que l'utilisateur ait un forfait comprenant un accès aux données. Ceci est une hypothèse forte qui peut restreindre l'adoption d'un tel système. Nous proposons donc d'ajouter un mode semi-connecté ainsi qu'un mode tout-déconnecté limité permettant de réaliser des transactions même sans aucune connectivité au réseau. Ces différentes solutions sont présentées et discutées dans le chapitre 3.

Après cette étude des architectures de sécurité existantes dans le domaine du

paiement mobile, un état de l'art des méthodes de fouilles de données fait l'objet de la prochaine section.

2.3 Algorithmes de classification

Tout d'abord, le problème de détection de fraudes et de classification est replacé dans le contexte plus global de l'extraction des connaissances. Ensuite, les différentes catégories d'algorithmes existants sont détaillées. Finalement, l'utilisation de ces algorithmes pour la détection de la fraude dans les paiements mobiles prépayés est discutée.

2.3.1 Positionnement du problème de détection de fraudes

Le processus d'Extraction de Connaissances à partir de Données, *ECD*, [FPSS96] décrit l'ensemble du cycle de découvertes de connaissances depuis la conception des bases de données aux traitements à effectuer pour extraire les connaissances à partir des données. La figure 2.4 montre le processus d'ECD. Tout d'abord, les données sont sélectionnées et nettoyées. Les doublons ou les données incomplètes sont par exemple retirées. Ensuite, les données sont transformées. On peut par exemple discrétiser des valeurs, calculer des moyennes, des intervalles de temps entre deux transactions ou d'autres informations issues de corrélation ou d'aggrégations. Ces informations sont les entrées des algorithmes de fouille de données. Les résultats obtenus durant cette phase doivent ensuite être interprétés pour extraire la connaissance des données de départ. Dans notre cas, la connaissance recherchée correspond à la labellisation d'une transaction en tant que transaction normale ou potentiellement frauduleuse.

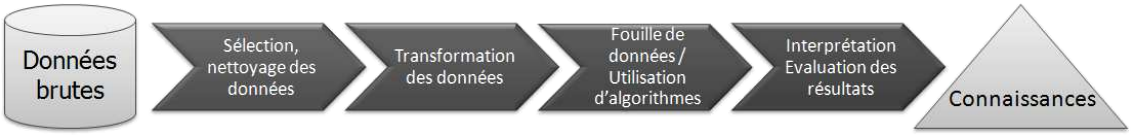


FIGURE 2.4 – Processus d'Extraction de Connaissances à partir de Données, adapté de [FPSS96]

Comme le montrent Fayyad *et coll.* [FPSS96], il existe différents problèmes de fouilles de données, tels que la *classification* qui consiste à créer une fonction qui labellise une donnée par rapport à des classes prédéfinies, la *régression* qui permet de créer une fonction capable de prédire une variable réelle à partir d'une donnée, le *clustering* qui permet de décrire les données en identifiant plusieurs catégories

qui les structurent ou le *résumé* qui vise à construire des représentations compactes de données. Cependant, comme le notent Fayyad *et coll.* [FPSS96], ces tâches sont basées sur des méthodes communes et peuvent également être utilisées les unes avec les autres. Il est par exemple possible d'utiliser des méthodes de résumé pour réduire la dimension des données, puis d'appliquer une méthode de clustering pour déterminer les classes à utiliser lors d'une phase de classification.

La détection de fraude peut donc être vue comme un problème de classification puisque l'objectif est de déterminer si une transaction ou un ensemble de transactions sont frauduleuses ou légitimes. L'objectif de la prochaine section est de présenter les différents types d'algorithmes de classification.

La figure 2.5 représente les différentes approches possibles de classification. Comme défini précédemment, cette tâche consiste à suivre un modèle pour attribuer à chaque donnée un label qui correspond à une catégorie prédéfinie. Ce modèle peut être prédéfini par un expert, être appris automatiquement ou être construit à partir de l'étude d'autres instances labellisées. Ces différentes catégories sont détaillées dans les prochaines sections et des exemples d'algorithmes sont décrits.

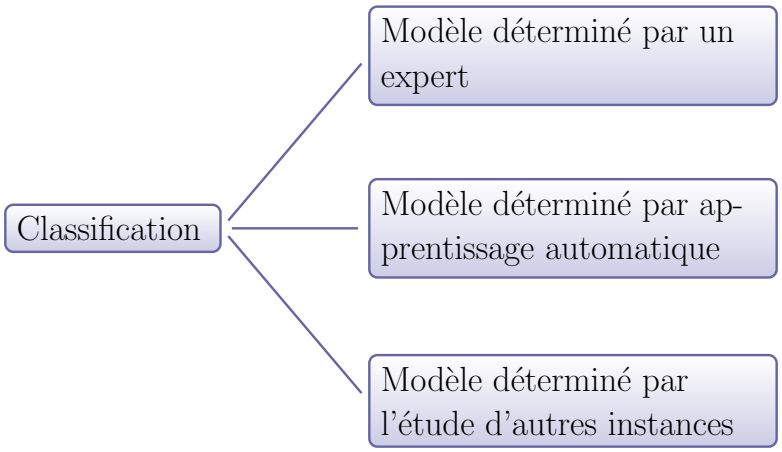


FIGURE 2.5 – Typologie des différentes formes de classification

2.3.2 Classification à partir d'un modèle prédéterminé par un expert

Dans ce cas la classification est réalisée en appliquant aux données un modèle basé sur des règles définies par un expert.

Règles

Des experts peuvent définir des règles qui sont utilisées pour classifier les données. Les règles implémentées correspondent le plus souvent à des règles métiers. Par exemple, si un porteur réalise plus de 10 transactions en moins d'une heure, la probabilité que ces transactions soient frauduleuses est élevée.

Ce processus ne correspond pas à une méthode de fouilles de données à proprement parler. Il est cependant possible d'utiliser la fouille de données pour trouver des règles et déterminer lesquelles sont les plus efficaces ou discriminatoires pour la détection de la fraude.

Systèmes experts

Les systèmes experts [Leo95] sont basés sur des règles définies par des experts. Ces dernières font partie d'une base de connaissances qui est associée à un moteur d'inférence. Ainsi, il est possible de reproduire le raisonnement d'un expert et de décider si une transaction est frauduleuse.

2.3.3 Classification basée sur l'apprentissage

Les méthodes d'apprentissage automatique utilisent les données en entrée pour construire un modèle qui sert à guider la classification. Le principe de la classification supervisée à base d'apprentissage est illustré par la figure 2.6. Celle-ci se déroule en deux phases. La première correspond à la construction d'un modèle mathématique permettant de décrire les données. La seconde phase consiste à appliquer ce modèle pour classifier d'autres données.

Il existe trois méthodes d'apprentissage automatique [PLSMG05] :

- l'apprentissage supervisé ;
- l'apprentissage non supervisé ;
- l'apprentissage semi-supervisé.

L'apprentissage supervisé consiste à donner au système un ensemble d'entrées $X_1, ...X_n$ et de sorties $Y_1, ...Y_n$ afin qu'il trouve des relations entre ces deux ensembles. Chaque X_i correspond à un vecteur représentant une transaction avec plusieurs variables descriptives $(x_1, ..., x_j, ..., x_m)$ telles qu'un montant ou une date par exemple. Chaque Y_i correspond au label, transaction légitime ou frauduleuse par exemple, associée au vecteur X_i. Le but de l"apprentissage supervisé est ensuite

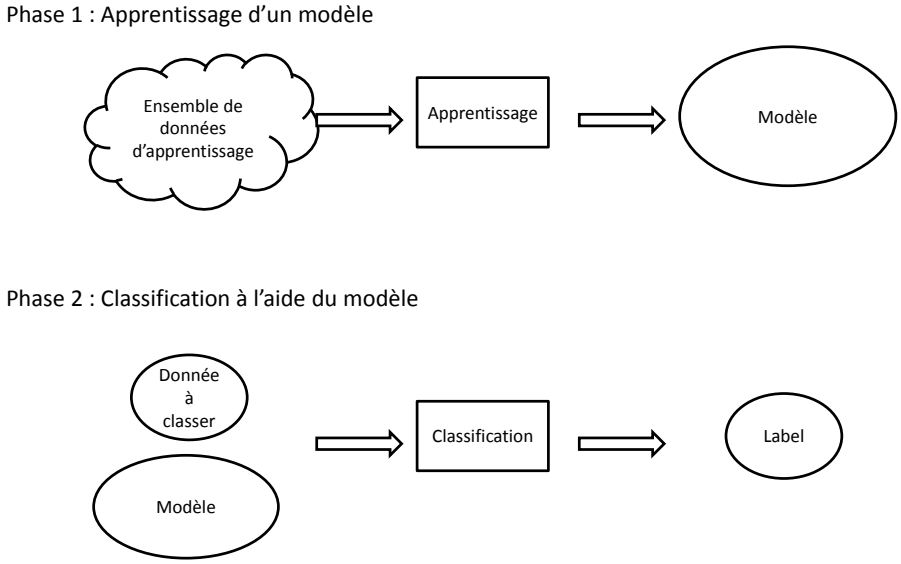


FIGURE 2.6 – Classification supervisée à base d'apprentissage

de prédire la sortie pour une entrée nouvelle. Dans le cas de la détection de la fraude, l'apprentissage supervisé consiste à observer des transactions (X_i) labellisées, c'est-à-dire pour lesquelles on dispose d'une information Y_i indiquant s'il s'agit d'une fraude ou non. Des relations entre les différents paramètres des transactions et les fraudes sont ensuite déterminées. Cela permet de classifier les nouvelles transactions.

Pour l'apprentissage non supervisé, le système reçoit uniquement les données d'entrée $X_1, ...X_n$ mais aucune autre indication. Son but est d'arriver à estimer le label Y_{n+1} de la prochaine entrée à partir de toutes les données qu'il a reçues précédemment. Cette méthode est entre autres utilisée pour détecter des données inattendues, inhabituelles ou aberrantes. Cette approche peut nous intéresser si on considère qu'une transaction frauduleuse fait partie des transactions anormales.

Dans le cas de l'apprentissage semi-supervisé, le système reçoit des entrées $X_1, ...X_n$ et un sous-ensemble de $Y_1, ...Y_n$. C'est une situation qui est possible si nous avons une base de données de transactions où certaines transactions sont labellisées en tant que fraude certaine, les autres transactions ne sont pas labellisées car il existe une incertitude quant à leur nature frauduleuse ou légitime.

Il est possible de combiner ces méthodes, c'est ce qu'on appelle les méthodes d'apprentissages hybrides. Généralement, deux méthodes supervisées ou plus sont utilisées conjointement ou une méthode non supervisée et une méthode supervisée.

Dans le cadre de cette thèse, seule la classification à base d'algorithmes d'apprentissage supervisé est considérée. Différents algorithmes de cette catégorie sont maintenant présentés.

Méthode statistique

La régression linéaire [WF05] est une méthode statistique de classification. Elle consiste à exprimer la valeur de la classe en fonction des instances sous la forme d'une équation linéraire de la forme $y = \alpha_1 x_1 + ... + \alpha_m x_m$ où y est une variable binaire qui représente l'appartenance à une classe, α_i représente des coefficients et x_i correspond aux différents variables qui permettent de décrire une donnée. Ensuite, les paramètres α_i sont optimisés pour prévoir au mieux la classe des données. Dans le contexte de la thèse, les valeurs x_i correspondent aux différents champs enregistrés pour décrire une transaction.

La régression logistique est une forme de régression linéaire pour laquelle on considère des probabilités d'appartenance à une classe plutôt qu'une valeur numérique qui représente la classe. Différentes méthodes existent pour réaliser une régression logistique. Par exemple, les arbres de régression logistique [LHF05] utilisent les arbres de décision pour diviser l'espace et appliquer une régression logistique à chaque partie. Ainsi, plusieurs équations de régression logistique au lieu d'une seule sont utilisées pour l'approximation. Les méthodes de régression logistiques multinomales [LCVH92] permettent de réaliser une régression logistique pour plusieurs classes. De plus, cette méthode introduit une contrainte, le paramètre *ridge* sur les coefficients de la régression logistique. Celui-ci permet une certaine tolérance sur les marges d'erreur de la régression logistique. Il permet ainsi d'éviter des problèmes de sur- ou sous-apprentissage. Appliqué à la régression linéaire, cela revient à minimimser $a_1 x_1 + ... + a_m x_m - y + ridge$

La classification naïve bayésienne [JL95] vise à estimer la probabilité d'appartenance à une classe en prenant en compte les probabilités conditionnelles qui lient les classes et les différentes variables caractéristiques des données. La relation de Bayes $P(A|B).P(B) = P(B|A).P(A)$ se traduit ici par $P(Y|x_1, ..., x_m).P(x_1, ..., x_m) = P(x_1, ..., x_j, ...x_m|Y).P(Y)$. Les différentes variables explicatives sont considérées comme indépendantes entre elles dans le cadre de la classification naïve bayésienne. La relation de Bayes devient donc $P(Y) = P(x_1, ..., x_j, ...x_m)$

Modèles de Markov cachés

Les modèles de Markov cachés [SKSM08] allient théorie des graphes et théorie des probabilités. Une chaîne de Markov cachée est un automate qui possède N états possibles $E_1, E_2, ..., E_N$. Le temps t est décomposé de manière discrète et se déroule pas à pas, $t = 0, 1, 2, 3....$ A l'instant $t = i$, le système se trouve dans un état particulier q_i, $q_i \in = E_1, E_2, ..., E_N$. A chaque incrémentation de t, l'automate change d'état suivant une certaine probabilité dépendant de l'état précédent.

En plus d'un automate tel que défini ci-dessus, un modèle de Markov caché est caractérisé par une matrice décrivant les probabilités de transition, un vecteur de probabilité de l'état initial et une séquence d'observations, une transaction dans notre contexte. Ces paramètres sont estimés à l'aide d'une phase d'apprentissage. Le but des chaînes de Markov est d'estimer si une transaction est frauduleuse à partir de la probabilité de l'observer dans une certaine séquence.

Réseaux bayésiens

Les réseaux bayésiens [Bou04] allient théorie des graphes et théorie des probabilités. Ils s'appuient sur la relation de Bayes $P(A|B).P(B) = P(B|A).P(A)$ qui permet de calculer des probabilités conditionnelles. Les réseaux bayésiens allient règles d'inférence et probabilités pour fournir une description des dépendances entre les variables décrivant une donnée. Si on connaît les dépendances entre le montant, le type de transaction et l'heure de transaction par exemple, il serait possible à partir de ces valeurs pour une transaction T donnée de déterminer si T est frauduleuse.

Réseaux de neurones

Un réseau de neurones est un modèle mathématique qui vise à reproduire le fonctionnement des neurones biologiques. Ce modèle permet de faire de l'apprentissage supervisé comme le perceptron ou non supervisé comme les cartes de Kohonen [Koh90]. Le réseau de neurones prend en entrée un vecteur $v = (x_1, ..x_n)$ auquel il doit attribuer une classe. Pour cela, différentes entités, les neurones, sont activées.

La figure 2.7 montre un neurone formel à la base des réseaux de neurones. Il est associé à une équation d'hyperplan qui sépare l'espace en deux classes et qui est caractérisé par les coefficients $w_1, w_2, ...w_n$. La phase d'apprentissage a pour but

d'optimiser ces valeurs et de trouver une équation séparant les classes au mieux.

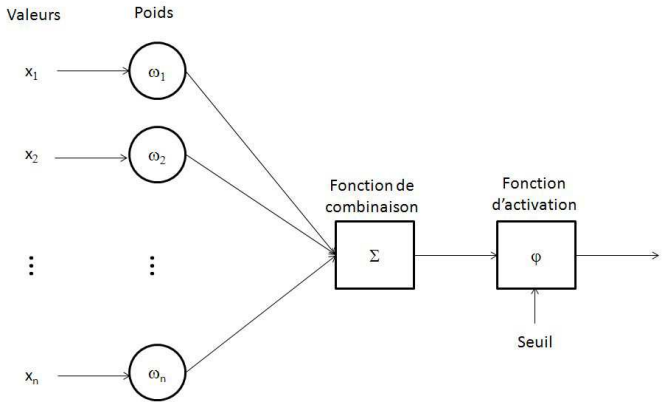


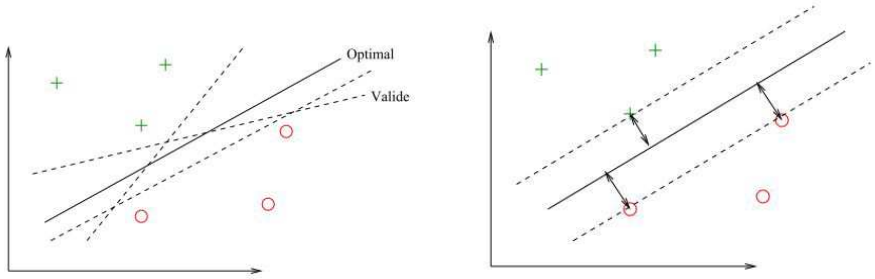
FIGURE 2.7 – Schéma d'un neurone formel, adapté de [WL90]

La séparation créée par un neurone est linéaire. Pour pouvoir séparer deux classes dans un cas non linéaire, il est possible de relier les neurones en réseau afin de créer des modèles plus complexes. Il existe d'autres types de réseaux de neurones mais ils ne sont pas détaillés ici.

Séparateurs à Vaste Marge

Les séparateurs à vaste marge [HDO+98] ou SVM ont pour objectif de trouver la meilleure frontière de décision pour séparer l'espace en deux régions. En cela, les SVM ressemblent aux réseaux de neurones. Ils sont cependant plus simples d'utilisation car ils dépendent entre autres d'une fonction noyau choisie par l'opérateur. De plus, ils ne nécessitent pas de préciser de structure contrairement aux raisons de neurones.

Les SVM se décomposent en deux étapes. Tout d'abord, les entrées sont transformées dans un espace F qui dispose d'un produit scalaire. Ensuite, on cherche à choisir une frontière optimale pour séparer les données des deux classes. La frontière est dite optimale si elle est le plus loin possible de tous les exemples. On va donc chercher à définir l'équation de l'hyperplan de séparation et à maximiser la distance du point le plus proche à l'hyperplan, c'est-à-dire la marge (c.f. figure 2.8). En pratique, l'utilisation de la fonction noyau permet de réaliser cela dans l'espace de départ.



(a) Recherche d'une frontière optimale (b) Marge à maximiser

FIGURE 2.8 – Principe de la recherche d'une frontière optimisée, adapté de [HDO+98]

Arbres de décision

Un arbre de décision regroupe un ensemble de règles qui permettent de classifier les données. Chaque nœud de l'arbre représente une règle sur une variable descriptive et un ET logique qui relie cette règle à celles des noeuds précédents. Une feuille de l'arbre correspond à une décision prise grâce à l'association des différentes règles menant à celle-ci. Chaque feuille est associée à un certain pourcentage de couverture pour un label donné.

L'algorithme ID3 [Qui93] permet de construire un tel arbre. A partir du noeud racine qui correspond à l'ensemble des données, cet algorithme sélectionne la meilleure variable descriptive ainsi qu'une valeur pour cette variable. Ensuite, l'algorithme divise les données en fonction de la valeur de la variable descriptive sélectionnée. Pour chaque sous-ensemble, l'algorithme choisit à nouveau la meilleure variable descriptive ainsi qu'une valeur pour diviser encore le sous-groupe et ainsi de suite jusqu'à ce qu'un sous-ensemble contienne uniquement des éléments d'une seule classe ou si l'ensemble des variables ont été considérées. L'algorithme C4.5 [Qui93] est une évolution d'ID3 qui prend en compte des plages de valeurs continues et qui procède à un élagage pour simplifier l'arbre en conservant ses performances.

Les forêts aléatoires [Bre01] sont constituées d'un certain nombre d'arbres de décision construits de manière partiellement aléatoire lors de la phase d'apprentissage [HBA+08]. Pour classer une entrée, celle-ci est d'abord classée par les arbres de la forêt. La classe qui lui est attribuée correspond à la classe déterminée par la majorité des arbres de la forêt.

Tables de décision

Les tables de décision [Koh95] sont constituées de deux parties. La première regroupe différentes variables et leurs valeurs possibles. La seconde contient une liste de transactions labellisées et leurs description en fonction des variables de la première partie. Pour classer une nouvelle instance, celle-ci est comparée aux instances existantes et est labellisée en fonction de cette comparaison. L'algorithme d'apprentissage permet de définir comment les variables descriptives sont choisies.

La table de décision majoritaire [Koh95] est construite en utilisant l'algorithme glouton *best-first* qui sélectionne les variables explicatives qui couvrent le plus de cas indépendemment les unes des autres.

La table de décision de type PART [FW98] est associée au classifieur C4.5. La table de décision est construite en plusieurs itérations. A chacune d'elles, un arbre de décision de type C4.5 est construit et la feuille qui couvre le plus de cas est sélectionnée. Les feuilles ainsi sélectionnées correspondent à un ensemble de règles qui permettent de construire la liste des variables de la table de décisions ainsi que ses différentes instances.

2.3.4 Modèle défini par l'étude d'instances

Ces algorithmes prennent en entrée une donnée à classer ainsi qu'un ensemble de données labellisées. Leur objectif est de trouver le label de l'entrée en le comparant aux autres instances.

La différence entre la classification à base d'apprentissage supervisé concerne la construction d'un modèle. Dans le cas de l'études d'instances, phase de construction d'un modèle n'existe pas. La donnée à classer est comparée avec les données existantes dont on connaît déjà la classe. Les données les plus proches déterminent la classe de la donnée en entrée de l'algorithme.

Les différents algorithmes qui forment cette catégorie se différencient par la fonction de calcul de distance. Celle-ci permet de déterminer de quelles données la donnée à classer est la plus proche.

Un exemple d'algorithme est la méthode des k-plus-proche voisins, [AKA91]. Elle se déroule en deux étapes. Tout d'abord, k est fixé et les k plus proches voisins de la

données à classer sont déterminés. Ensuite, l'étiquette la plus représentée parmi les k plus proches voisins est choisie pour la donnée à classer. La figure 2.9 représente la méthode pour $k = 3$ dans un espace à deux dimensions. Les données labellisées sont réparties en deux classes différentes, la classe bleue et la classe rouge. Après calcul des distances entre la donnée à classer et les autres données, il apparaît que les 3 plus proches voisins comprennent deux données bleues et une rouge. La donnée appartient donc à la classe bleue.

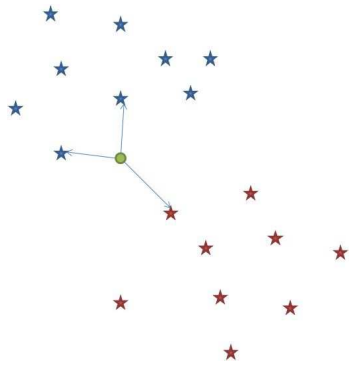


FIGURE 2.9 – Méthode des k-plus-proches voisins

Dans l'exemple précédent, la distance euclidienne est considérée. D'autres distances peuvent être utilisées comme les distances de Manhattan ou de Tchebychev. Il est également possible d'utiliser des méthodes qui ne se basent pas sur la localisation mais sur le coût nécessaire pour transformer une donnée en une autre. C'est le cas de la méthode K* [CT95].

2.3.5 Discussion sur les algorithmes de classification

L'état de l'art réalisé ici regroupe plusieurs méthodes de fouilles de données mais ne peut pas être exhaustif tant ce domaine est étendu. Les méthodes présentées sont complémentaires et peuvent être utilisées les unes avec les autres. Les algorithmes ont également des caractéristiques différentes qui peuvent être exploitées de plusieurs manières selon l'utilisation que l'on veut en faire. Phua *et coll.* [PLSMG05] réalisent un état de l'art des méthodes de classification utilisées pour détecter la fraude dans différents domaines, la détection de terrorisme, de crimes financiers, d'intrusion et de spams. Ces travaux montrent que la détection dans ces domaines est réalisée de manière différente même si certaines méthodes sont communes aux différents domaines. Résoudre un problème avec des algorithmes de fouilles de données requiert donc de

spécifier des objectifs clairs et d'adapter l'utilisation des algorithmes au problème visé.

Les paiements bancaires utilisent des outils de détection et de gestion de fraudes depuis de nombreuses années. Par exemple, Visa et Mastercard proposent des solutions de détection de la fraude, respectivement *Real Time Scoring*, RTS [VIS] et *Risk Finder* [Mas]. Cependant, les paiements mobiles prépayés, comme tout nouveau produit génèrent de nouveaux risques. Les usages des utilisateurs de ces deux services et les technologies mises en oeuvre ne sont pas les mêmes. En conséquence, même si des outils de fouilles de données identiques sont utilisés dans ces deux domaines, il est nécessaire de les adapter au contexte des paiements sur mobile prépayés.

De nombreuses méthodes citées ci-dessus sont appliquées dans le domaine bancaire. A notre connaissance, les méthodes les plus utilisées sont les filtres, les arbres de décision ou des arbres de décision associés à une régression logistique, forme particulière de régression linéaire. Ces méthodes sont souvent préférées car elles facilitent l'interprétation et la visualisation. Il est alors plus facile d'expliquer au client pourquoi une transaction est soupçonnée d'être frauduleuse. D'autres méthodes plus complexes sont également mises en oeuvre. Par exemple, VISA utilise les réseaux neuronaux dans leur service de lutte contre la fraude RST, *Real Time Scoring* [VIS]. Cet outil associe un score à chaque transaction et lève une alerte si le score dépasse un certain seuil défini par les banques qui s'en servent. Cet outil permet de motiver un blocage mais la banque qui l'utilise a la responsabilité de trouver les variables explicatives pour interpréter le blocage.

Bhattacharyya *et coll.*, Delamaire *et coll.* et Al-Khatib [BJTW11, DAP09, AK12] réalisent un état de l'art des différents algorithmes de fouille de données utilisés pour la détection de la fraude dans le domaine des paiements par carte. Nous retrouvons ainsi que différents travaux ont été menés pour adapter les réseaux neuronaux, les arbres de décision, les systèmes experts, les modèles de Markov cachés, les SVM, les réseaux bayésiens à la détection de la fraude pour les transactions réalisées avec la carte bancaire.

A notre connaissance, tous les services de paiement sur mobile n'implémentent pas des solutions de gestion de fraude automatisées. Parfois la surveillance des comptes se fait de manière manuelle ou avec des outils de détection basés sur des règles métiers. Le service M-Pesa a déployé Minotaur^{TM} Fraud Management Solution en 2012 [Oku12]. Ce système de gestion de la fraude est basé sur l'utilisation de règles

métiers et de réseaux de neurones [Neu]. A notre connaissance, il n'existe pas de travaux publics concernant l'étude et l'adaptation de méthodes de détection de fraude aux systèmes de paiement sur mobile.

2.4 Discussion

Comme nous l'avons vu dans le chapitre 1, cette thèse s'inscrit dans le contexte des services de transaction sur mobiles. Ces services sont aujourd'hui basés sur le canal USSD qui n'est pas prévu aujourd'hui dans les réseaux tout-IP. Ces services prennent comme hypothèse que le réseau qu'ils traversent et les terminaux permettant de réaliser les transactions sont sûrs. Cette hypothèse est remise en question par l'exploitation d'une faiblesse des réseaux 2G [NP09] et par l'utilisation croissante des *smartphones*. Ceux-ci sont des « téléphones intelligents » et ont accès à l'internet. Ils sont donc aussi vulnérables que les ordinateurs aux virus et chevaux de Troie. Il est donc nécessaire d'anticiper l'évolution des services de paiement prépayés mobiles vers l'utilisation des réseaux LTE et des smartphones. De cette manière, les vulnérabilités connues sont abordées en tirant avantage des nouvelles fonctionnalités fournies par ces nouvelles technologies.

La problématique de cette thèse est la sécurisation des transactions sur terminaux mobiles. A partir du cycle de gestion de la fraude présenté dans le chapitre 1, nous avons choisi de nous intéresser aux architectures de sécurité existantes dans ce domaine et aux algorithmes permettant d'analyser les journaux d'événements automatiquement. Dans ce chapitre 2, nous avons réalisé un état de l'art des architectures de sécurité dans le domaine du paiement mobile ainsi qu'un état de l'art des algorithmes de fouilles de données. Ces deux études justifient les deux axes que nous avons choisis pour développer cette thèse.

En effet, il n'existe pas d'architecture de sécurité adaptée au contexte et aux contraintes que nous avons présentés. Il existe de nombreux algorithmes de fouille de données permettant de faire de la détection de fraude. Ceux-ci réalisent des objectifs différents et ciblent des données diverses. Il est donc nécessaire de faire des choix pour adapter ces algorithmes au problème à traiter. Il n'est pas possible de transposer directement les travaux réalisés pour la détection de la fraude dans les paiements par carte bancaire au domaine des paiements mobiles prépayés.

La suite de ce manuscrit détaille les contributions de cette thèse dans ces deux

domaines. Le chapitre 3 décrit l'architecture et les protocoles de paiement que nous proposons. Le chapitre 4 expose quant à lui les contributions dans le domaine de la détection de la fraude pour les paiements mobiles prépayés.

Chapitre 3

Architecture de confiance pour les services de transactions sur terminaux mobiles

Dans ce chapitre, nous étudions différents types de transactions et proposons des protocoles sécurisés de bout-en-bout pour chacune d'entre elles. Pour cela, nous détaillons pour chaque type de transaction les besoins de sécurité en examinant les flux bruts qui constituent cette forme de transaction. Ensuite, les protocoles proposés sont décrits et validés en montrant que les propriétés de sécurité souhaitées sont bien respectées.

Sommaire

3.1 Présentation des protocoles

Dans cette première section, une architecture et différents protocoles sont proposés pour garantir une sécurité de bout-en-bout entre l'application et la plateforme de paiement.

TABLE 3.1: Symboles utilisés

X	Entité X =P, M, Pp, D ou E	$hash()$	Fonction de hachage
P, M, D ou E	Entité porteur, marchand, destinataire ou expéditeur	Pp	Entité plateforme de paiement
PK_X	Clé publique de X	pk_X	Clé privée de X
$\{m\}_{PK_X}$	Message m chiffré par PK_X	ID_X	Identité de X
Not	Notification, indique si la transaction a réussi ou échoué	$Alea$	Nombre aléatoire
TD	Date de la transaction	TT	Type de la transaction
TA	Montant de la transaction	.	Opérateur de concaténation
$a.b.c.SIG_X$	Signature par X du haché de a.b.c	$SIG_X(m)$	Sigature par X du haché de m
$PreBal_X$	Solde de X avant la transaction	$PostBal_X$	Solde de X après la transaction
HDR	Entête des messages, dépend du protocole utilisé	SK_{XY}	Clé secrète partagée par X et Y
PMK	Clé pré-maître	$CERT(X)$	Certificat de X
$Verif_X$	Valeur de la vérification du code secret de X	$Auto_X$	Autorisation d'un paiement hors ligne par l'entité X
Req_z ou Rep_z ou Ack_z	Requête, réponse ou Acquitement de z=Val, Tran, ID, Proposition, Choix, Défi, Contrat ou Maj	Val	Validation
$Tran$	Transaction	[a]	Élément optionnel
$Contrat$	Contrat	Maj	Mise à jour
CP_{TID}	Contrat de paiement de la transaction TID	$CP_{1..n}$	Ensemble des n contrats de paiement de 1 à n
TID_{XY}	Numéro d'identification de la transaction générée entre les entités X et Y	TID_X	Numéro d'identification de la transaction générée par l'entité X
TID	Numéro d'identification de la transaction	Nb_C	Nombre de contrats
KE_X	Champ contenant l'échange de clés Diffie-Hellmann émis par l'entité x, défini dans la norme RFC 5996 [IETa]	SA_X	Association de sécurité émise par l'entité X, définie dans la norme RFC 5996[IETa]
$Auth_X$	Champ émis par X dans le protocole IKEv2 permettant l'authentification mutuelle, défini dans la RFC 5996 [IETa]	SA_{XY}	Association de sécurité négociée par X et Y

3.1.1 Architecture, hypothèses et notations

Les différents notations et symboles utilisés dans ce chapitre sont regroupés dans le tableau 3.1. Les transactions sont réalisées à l'aide de m-monnaie, notée m. Celle-ci représente la monnaie électronique émise par l'opérateur qui gère le service de transactions sur terminaux mobiles.

Nous supposons que les systèmes d'exploitation déployés de base dans les téléphones ne sont pas de confiance. Pour établir la sécurité dans les téléphones, nous utilisons ici conjointement un élément sécurisé, SE, et un environnement d'exécution

sécurisé, TEE. L'élément sécurisé permet de gérer les secrets et les opérations les plus critiques de notre architecture. L'environnement d'exécution sécurisé, TEE, permet un accès sécurisé aux périphériques du téléphone, clavier, écran, mémoire et autres. Il permet également d'avoir un environnement moins restreint et plus performant que l'élément sécurisé. Ainsi, des opérations moins critiques et de complexité importante peuvent être exécutées ici.

Nous souhaitons utiliser la carte SIM en tant qu'élément sécurisé puisqu'il s'agit de l'élément le plus répandu. De plus, l'opérateur possède la SIM et gère le système de transfert d'argent sur mobile. Notre architecture est donc basée sur une application dans la carte SIM, une application dans l'environnement d'exécution sécurisée et une application dans la plateforme de paiement. Dans la suite, nous parlons simplement de SIM, de TEE et de plateforme pour évoquer ces applications.

Chaque acteur, porteur, marchand ou agent souscrit au service de transactions sur mobile. Chacun dispose d'une paire de clés publique, privée, ainsi que d'un certificat stockés dans sa carte SIM. La plateforme de paiement dispose également d'une paire de clé publique, privée. Comme l'opérateur gère à la fois la plateforme, l'infrastructure de clés publiques et les cartes SIM, nous supposons que le certificat de la plateforme est déployé dans les cartes SIM des porteurs et des marchands. Le certicat comprend la clé publique de la plateforme. Ce certificat est auto-signé par la plateforme. Compte tenu que le modèle de paiement est un modèle trois-coins, l'opérateur gère tous les souscrivants du service. Une tierce partie n'est donc pas nécessaire pour garantir la confiance des certificats. Ils sont signés par la plateforme qui représente l'opérateur.

Les problématiques liées à l'interruption des échanges à cause d'une panne ou d'une interruption volontaire des canaux de communication ne sont pas pris en compte dans ces travaux.

Dans ce chapitre, nous ne considérons que les transactions de base comme le paiement en face à face (les retraits et les débits sont compris dans cette définition) et les transferts entre particuliers. Pour les transferts entre particuliers, nous considérons que ceux-ci ne sont pas sur le même lieu physique. Une exception, présentée en section 3.1.6, est faite pour le mode tout-déconnecté où les porteurs doivent être à proximité pour réaliser la transaction. Des transactions autres que celles de base peuvent exister, comme le paiement de factures par exemple, mais nous considérons que ces transactions dérivent des transactions de base. Nous considérons également

que les opérateurs n'ont pas le droit d'accorder des crédits aux utilisateurs du service de transaction sur mobile.

Ci-dessous, lorsque nous décrivons les flux, nous utiliserons le terme appareil pour désigner indistinctement le terminal mobile, la carte SIM ou l'environnement d'exécution sécurisé. Le choix d'attribuer un traitement à un de ces composants se fait dans la description détaillée des protocoles.

3.1.2 Canal sécurisé

Certains protocoles proposés s'appuient sur un canal sécurisé et d'une clé partagée entre deux entités. Le protocole IKEv2 [IETa] nous permet de créer cette clé partagée qui est utilisée pour chiffrer les échanges au niveau applicatif.

Ce protocole est constitué de deux phases illustrées en figure 3.1. Dans la première, a lieu un échange Diffie Hellmann pour calculer une clé de session. La seconde phase est composée de deux échanges où les deux parties s'authentifient mutuellement en échangeant leur certificat respectif ainsi que la valeur $AUTH_X$ ou $AUTH_{Pp}$. Ce champs défini dans la RFC [IETa] correspond au haché des deux messages échangés précédemment signé avec leur clé privée respective. La vérification de cette signature permet d'authentifier chacune des deux parties. Les messages échangés dans cette deuxième phase sont chiffrés avec la clé secrète partagée, ce qui permet d'assurer l'anonymat des entités en jeu. Comme nous supposons que le certificat de la plate-forme est déjà présent dans les éléments sécurisés des utilisateurs, son envoi dans le quatrième message est optionnel.

Les différents messages exposés ici comprennent une entête HDR définie dans le standard concernant le protocole IKEv2[IETa]. Les messages échangés par la suite sont sécurisés par un nouveau protocole ESP (Encapsulating Security Payload) défini dans le standard [IETe]. Celui-ci s'appuie sur des clés établies par IKEv2 entre les deux entités. Ils comprennent également une entête HDR définie dans le standard [IETe].

La clé de session produite dans ce protocole peut être rafraîchie de la manière décrite dans le protocole IKEv2. Une politique de sécurité définit les modalités de ce rafraichissement. Par exemple, la clé de session peut être valable pour une durée ou pour une quantité de données particulière. Au-delà, une nouvelle clé de session est recalculée. On peut également définir le nombre de clés de sessions

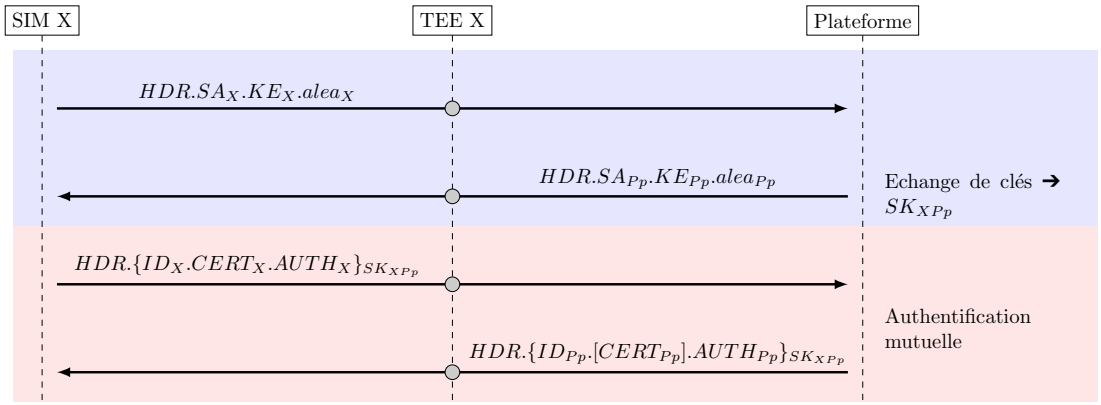


FIGURE 3.1 – Etablissement d'un canal sécurisé IKEv2 adapté à notre contexte pour mettre en place un canal sécurisé au niveau applicatif

recalculées. L'environnement d'exécution sécurisée relaie les différents messages entre la plateforme et la SIM. Il se charge d'extraire les données des paquets IP provenant de la plateforme et de les traduire en messages APDU [ISO], Application Protocol Data Unit, permettant de communiquer avec la SIM. Cette opération est répétée dans l'autre sens pour transmettre les messages de la SIM vers la plateforme.

3.1.3 Transfert entre particuliers

Avant de décrire le protocole que nous proposons afin de réaliser des transferts entre particuliers, nous décrivons ici les flux qui composent ce type de transaction ainsi que les besoins de sécurité associés. Les différents flux sont indicatifs et peuvent conduire à l'échange de plusieurs messages dans le protocole proposé.

Flux

Comme illustré en figure 3.2, afin de réaliser un transfert à un autre particulier, l'expéditeur initie tout d'abord la transaction. Il envoie alors une requête de transaction à la plateforme. Cet échange doit permettre à la plateforme de connaître tous les détails de la transaction et en particulier d'identifier l'expéditeur et le destinataire. La plateforme s'assure ensuite que l'expéditeur est l'utilisateur légitime du service. Ce dernier besoin est plus fort encore pour l'expéditeur, pour éviter que la palteforme ne traite des demandes de porteurs non légitimes.

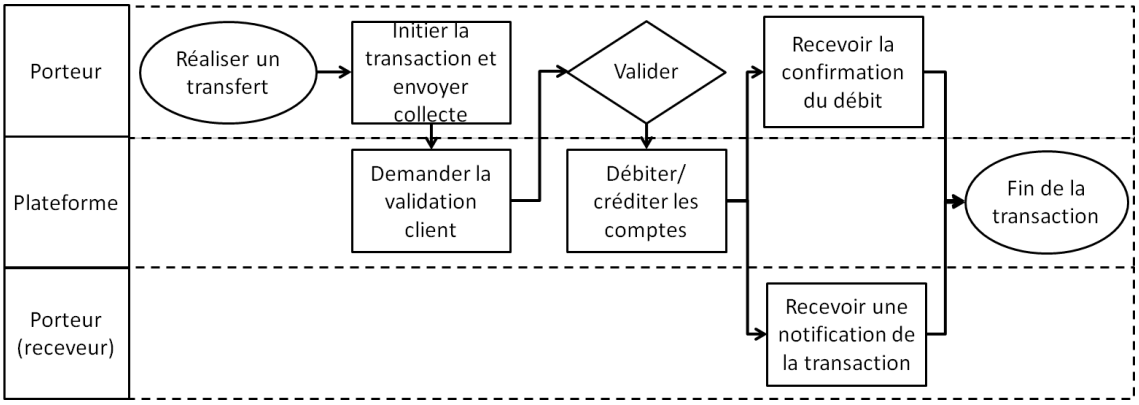


FIGURE 3.2 – Flux correspondant au transfert entre particuliers

La plateforme peut alors créer une nouvelle transaction et demander la validation à l'expéditeur. Ce flux permet à celui-ci de s'assurer que la transaction traitée par la plateforme est conforme à ce qu'il s'attend. Il se doit donc d'être intègre et provenir effectivement de la plateforme de paiement. Après que l'expéditeur valide la transaction, la plateforme de paiement peut débiter et créditer les comptes. Pour cela, la plateforme doit avoir une preuve que le porteur qui effectue le transfert a lui-même validé la transaction. De plus, celui-ci ne doit pas pouvoir contester être à l'origine de cette preuve.

Finalement, l'expéditeur et le destinataire sont informés de la transaction. Cette information sert de preuve que le transfert a bien eu lieu et a bien été traité par la plateforme de paiement. Finalement, les informations échangées doivent être confidentielles. En particulier, le protocole proposé doit garantir l'anonymat des deux porteurs vis-à-vis de tout acteur externe à la transaction. La notification du porteur destinataire est uniquement informative, et nécessite juste de provenir d'une plateforme légitime. En résumé, les différents besoins de sécurité pour cette transaction sont les suivants :

- confidentialité et intégrité des échanges ;
- anonymat des porteurs vis-à-vis des acteurs externes à la transaction ;
- preuve que l'expéditeur a bien validé la transaction en cours, ce qui implique une non-répudiation et une protection contre le rejeu de la validation ;
- preuve que la plateforme a bien validé la transaction, ce qui implique la non-répudiation et la protection contre le rejeu de la confirmation de la transaction à l'expéditeur ;
- authenticité de la notification de la transaction au destinataire ;
- identification des acteurs de la transaction ;

– authentification mutuelle de la plateforme et de l'expéditeur ;
– authenticité et intégrité des messages échangés.

Protocole

Le protocole proposé pour le transfert d'argent entre particuliers est représenté en figure 3.3. Tout d'abord, l'expéditeur initie la demande de paiement en saisissant le montant et l'identité de son destinataire. Sa carte SIM établit alors un canal sécurisé avec la plateforme de paiement et calcule la clé de session SK_{PPp} qui est utilisée pour chiffrer les prochains messages. Comme nous l'avons vu dans la partie 3.1.2, ce protocole assure également une authentification mutuelle. La carte SIM du porteur envoie ensuite la requête de transaction contenant le montant, l'identité de l'expéditeur, celle du destinataire, un aléa permettant de saler le message. Une signature de ces éléments par le porteur est jointe au message. L'identité permet à la plateforme d'identifier de manière unique l'expéditeur et le destinataire. Il peut s'agir de leur numéro de téléphone par exemple.

A la réception de ce message, la plateforme crée un identifiant de transaction qui identifie la transaction de manière unique et qui comprend une indication sur la date de la transaction. La plateforme demande ensuite à la SIM du client de valider cette transaction et lui fournit un défi $alea_2$ dont la validation devra dépendre. Cette requête est accompagnée d'un récapitulatif et de l'état du compte du porteur.

La carte SIM fait afficher par l'environnement d'exécution sécurisée ces éléments au porteur afin d'obtenir sa validation. Dans ce cas, celle-ci se manifeste par la saisie d'un code spécifique à l'application. Ce code est vérifié par la carte SIM qui envoie ensuite une preuve de la validation à la plateforme. Celle-ci est constituée de la valeur $Verif_P$ qui indique le succès ou l'échec de la validation par le porteur et de $SIG_P(Verif_P, alea_2)$. Cet élément permet de vérifier l'intégrité et l'authenticité de la validation de l'expéditeur puisqu'il s'agit du résultat d'une fonction de hachage, donc à sens unique et sans collision et du chiffrement par la clé privée de l'expéditeur qu'il est le seul à détenir. Le non-rejeu est garanti par l'utilisation du défi envoyé par le serveur dans l'échange précédent.

L'expéditeur envoie à son tour le défi $alea_3$ qui permettra au serveur de construire la preuve de paiement de la même manière que précédemment. Cette preuve est envoyée comme notification de la réussite ou de l'échec du paiement au client. Elle peut éventuellement être stockée, par exemple sur la plateforme de paiement ou dans

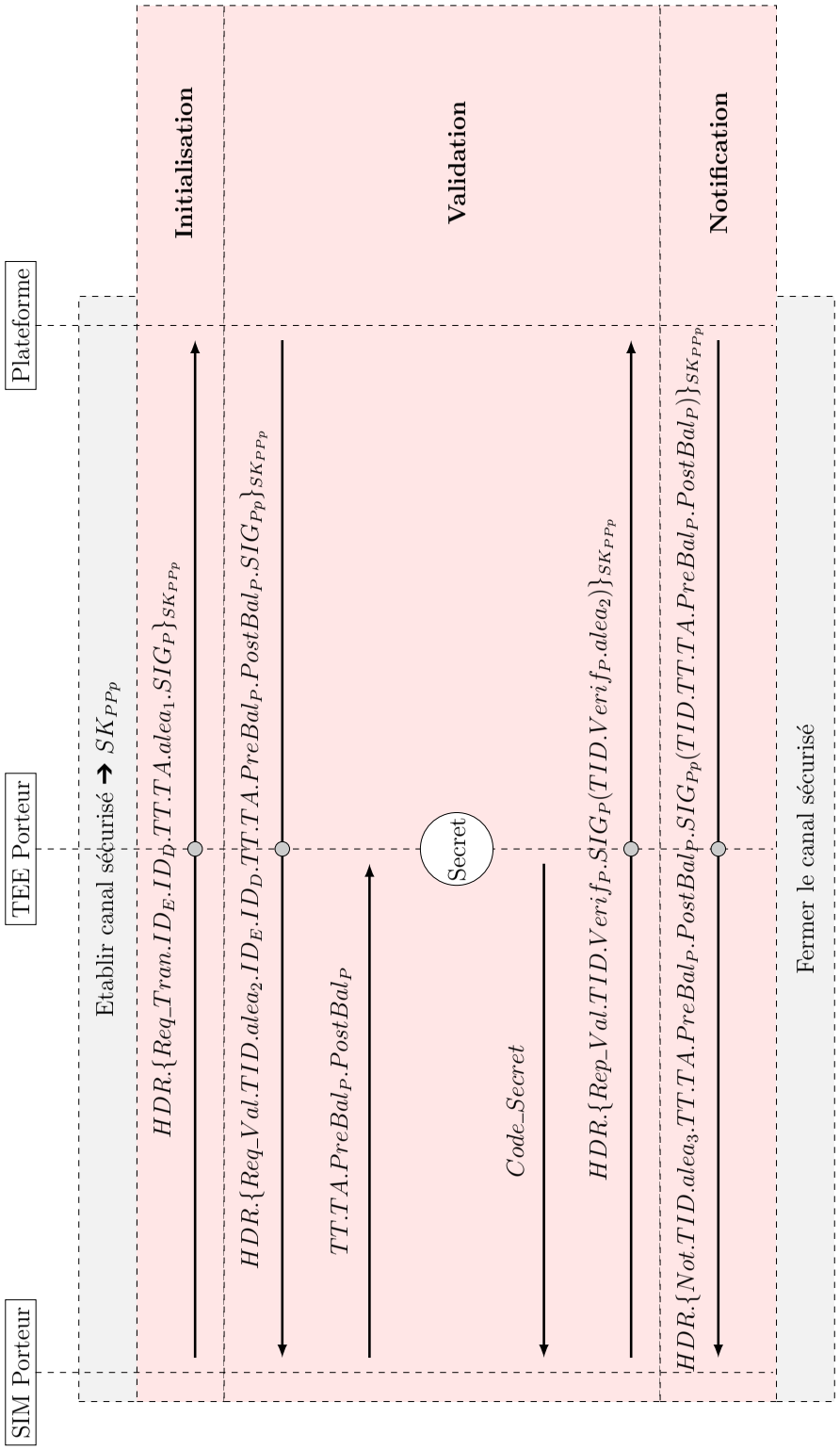


FIGURE 3.3 – Protocole pour le transfert C2C

l'environnement d'exécution sécurisé. L'anonymat des porteurs est garanti par la confidentialité des échanges. La présence d'une signature dans chacun des messages permet de garantir l'intégrité et l'authenticité de chacun des messages.

3.1.4 Paiement de proximité en mode tout-connecté

Ce type de transaction correspond à un paiement marchand, un retrait ou un dépôt d'argent. Cette transaction peut donc donner lieu à un transfert d'un marchand vers un porteur. C'est le cas du dépôt où le marchand vend de la m-monnaie qu'il transfert sur le compte du porteur. Elle peut également résulter en un transfert d'un porteur vers un marchand. C'est le cas lorsque le marchand vend un bien au porteur ou lorsque le porteur retire de l'argent en vendant de la m-monnaie au marchand.

Flux

Les différents flux constituant un paiement de proximité en mode tout-connecté sont représentés en figure 3.4. Ces flux s'appliquent dans les cas détaillés ci-dessus. Afin de réaliser un tel paiement, le porteur négocie le paiement avec le marchand qui initie alors la transaction. Il envoie alors une requête de transaction à la plateforme. Comme pour le transfert entre particuliers, la requête de transaction permet à la plateforme de connaître le montant de la transaction, d'identifier l'expéditeur et le destinataire, et de s'assurer que les acteurs de la transaction sont des acteurs légitimes. La plateforme peut alors demander la validation au porteur. Comme pour le transfert entre particuliers, ce flux permet au porteur de s'assurer que la transaction traitée par la plateforme est conforme à ce qu'il attend. Il se doit donc d'être intègre et provenir effectivement de la plateforme de paiement.

Après que le porteur valide la transaction, la plateforme demande également la validation du marchand. Cette deuxième validation est nécessaire car, pour l'opération de dépôt, le compte en m-monnaie du marchand est débité. Ensuite, la plateforme de paiement peut débiter et créditer les comptes. Comme précédemment, le porteur et le marchand ne doivent pas pouvoir répudier la transaction à laquelle ils ont participé. Finalement, chacune des parties de la transaction est informée de la réussite ou de l'échec de celle-ci et reçoit une preuve que la transaction a eu lieu et a été traitée par la plateforme de paiement.

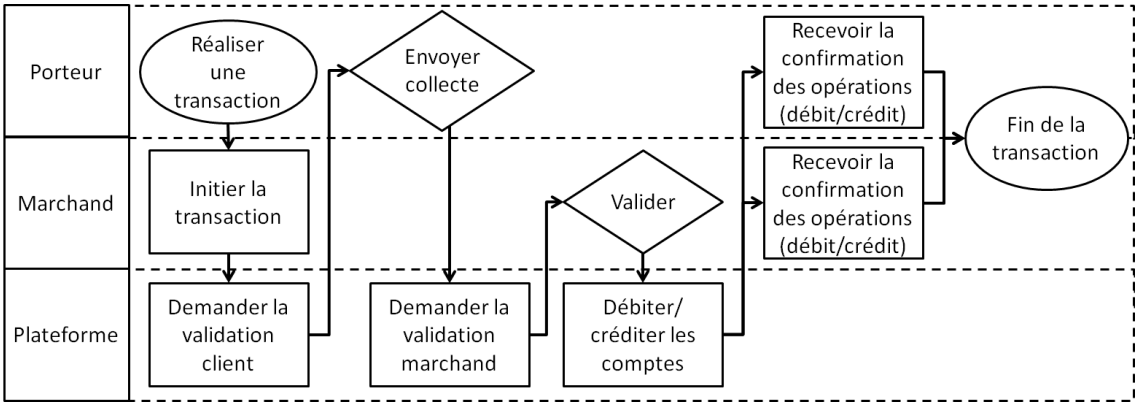


FIGURE 3.4 – Flux correspondant au paiement de proximité en mode tout-connecté

Finalement, les informations échangées ci-dessous doivent être confidentielles. En particulier, le protocole proposé doit garantir l'anonymat du porteur vis-à-vis du marchand et de tout acteur externe à la transaction.

En résumé, les différents besoins de sécurité pour cette transaction sont les suivants :

- confidentialité et intégrité des échanges ;
- anonymat du porteur vis-à-vis des acteurs externes à la transaction et vis-à du marchand ;
- non-répudiation de la validation par le porteur et le marchand ;
- non-répudiation de la confirmation de la transaction par la plateforme de paiement ;
- identification des acteurs de la transaction ;
- authentification mutuelle de la plateforme et du marchand ;
- authentification mutuelle de la plateforme et du porteur ;
- authenticité et intégrité des messages échangés.

Protocole

Le protocole proposé pour réaliser une transaction en face à face en mode connecté est représenté en figure 3.5. Le marchand initie la transaction en saisissant un montant. Sa carte SIM établit alors un canal sécurisé et une clé SK_{PpM} partagée avec la plateforme de paiement. Elle envoie alors une requête de transaction à la plateforme en indiquant l'identité du marchand, le montant et un identifiant de la transaction temporaire propre au marchand. La plateforme de paiement crée alors une transaction dans sa base de données et un numéro d'identification de la transaction. La plateforme répond avec une requête d'identification du client qui rappelle l'identifiant

de transaction temporaire, un cookie, et la signature par la plateforme de celui-ci. Le cookie est un élément qui ne peut être créé que par la plateforme de paiement et qui n'est compréhensible que par celle-ci. Le cookie identifie de manière unique la transaction. Il comprend également une date, ce qui permet à la plateforme de rejeter des cookies trop anciens.

Nous proposons que le cookie contienne deux parties. La première correspond à l'identifiant de transaction, du montant, de l'identifiant du marchand, sa date de fin de validité t_{valid} chiffrés par la clé publique de la plateforme de paiement. La seconde partie, qui permet de vérifier l'intégrité et l'authenticité de la première partie correspond à la signature de la première partie. La première partie du cookie est confidentielle et ne peut être interprétée que par la plateforme. Cependant, elle peut être créée par toute entité qui possède la clé publique de la plateforme alors que ce n'est pas le cas de la seconde partie du cookie. Celle-ci permet de garantir l'origine du cookie sans dévoiler son contenu. Le cookie correspondant est par exemple : $Cookie = \{TID.IDm.TA.t_{valid}.alea\}_{Pk_{Pp}}.\{h(\{TID.IDm.TA.t_{valid}.alea\}_{Pk_{Pp}})\}_{pk_{Pp}}$.

Le marchand s'assure que le cookie provient bien du serveur et le transmet avec sa signature au porteur. Cette transmission se fait par un moyen de communication de proximité tel que le NFC (Near Field Communication) ou un canal de communication hors-bande tel qu'un QRcode. Le client s'assure à son tour que le cookie provient bien de la plateforme de paiement à l'aide de la signature et établit à son tour un canal sécurisé avec la plateforme. La carte SIM du porteur renvoie alors à la plateforme le cookie, son identité et la signature de son identité. La plateforme de paiement s'assure ensuite que le cookie correspond bien à une transaction en cours, que la date de validité du cookie n'est pas dépassée, et que l'identité du client correspond bien à la signature produite.

L'ensemble des échanges impliquant le cookie permettent à la plateforme d'identifier le porteur à l'origine de la transaction sans jamais dévoiler son identité au marchand. La plateforme complète ensuite la transaction créée avec l'identité du client et envoie une demande de validation au marchand et au client. La validation et la notification se déroulent pour le porteur et le marchand de la même manière que dans le protocole de transfert d'argent entre particuliers. Les mêmes besoins de sécurité sont donc satisfaits.

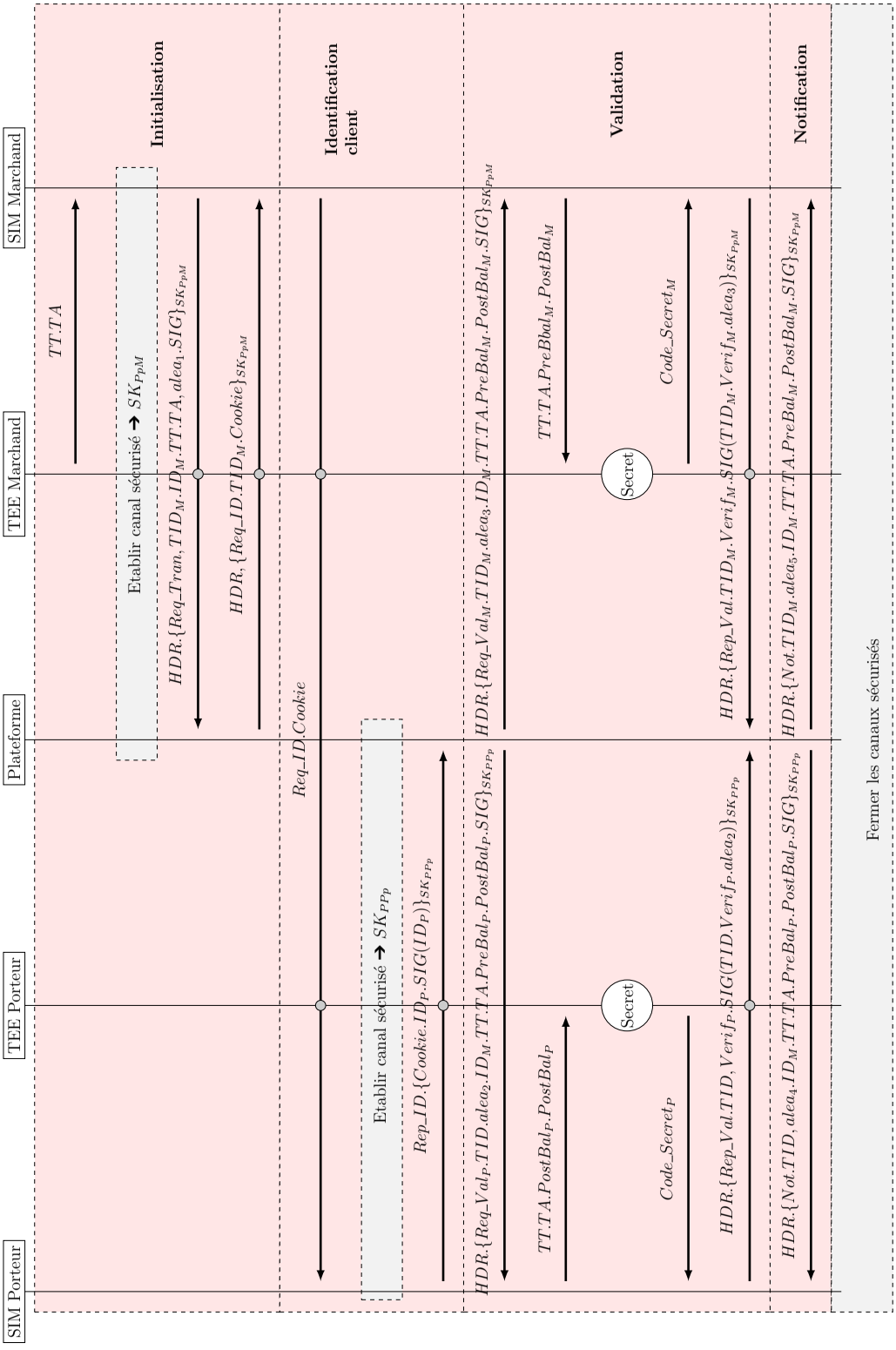


FIGURE 3.5 – Protocole pour le paiement de proximité en mode tout-connecté

3.1.5 Paiement de proximité en mode semi-connecté

En mode semi-connecté, nous supposons qu'un des acteurs n'a pas la possibilité d'interagir avec la plateforme de paiement. Cela correspond au cas où un porteur n'a pas de forfait permettant d'avoir accès aux données. Nous estimons que si un marchand se trouve dans une zone couverte, les accords le liant à l'opérateur qui gère la plateforme de paiement incluent un accès aux données pour le marchand. Nous proposons donc de traiter ce cas en faisant profiter au porteur l'accès du marchand aux données, ce qui permet ensuite de réaliser une transaction dans le même cadre que le mode tout-connecté.

Flux

Les différents flux de ce mode sont représentés en figure 3.6. Tout d'abord, le porteur qui souhaite réaliser une transaction initie le partage de connexion. Le marchand vérifie ensuite que le porteur est légitime et donc que la connexion peut être partagée avec lui. Si c'est le cas, le marchand partage sa connexion et la transaction peut se réaliser en mode connecté. Ainsi, les besoins de sécurité de ce mode sont :

- anonymat du porteur vis-à-vis du marchand ;
- accès au partage de connexion réservé aux clients du service ;
- accès au réseau virtuel et partage d'accès à ce réseau réservé aux marchands partenaires de l'opérateur de téléphonie sur mobile.

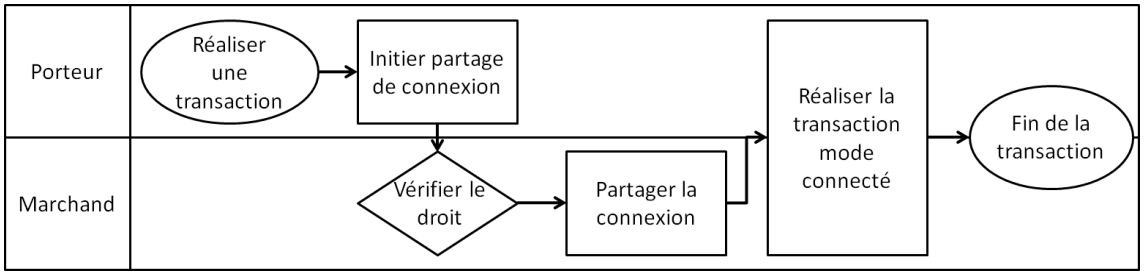


FIGURE 3.6 – Flux correspondant au paiement de proximité en mode semi-connecté

Protocole

Afin de faire bénéficier au porteur l'accès aux données du marchand, nous nous basons sur les protocoles de partage de connexion déjà mis en œuvre dans le cadre des réseaux wifi : une variante du protocole EAP-TLS, Extensible Authentication Protocol - Transport Layer Security [IETd], le protocole EAP-TTLS, Extensible

Authentication Protocol Tunneled Transport Layer Security [IETf]. Ce protocole permet une authentification mutuelle de la plateforme de paiement et du porteur qui souhaite bénéficier de la connexion tout en assurant la confidentialité de l'identité du porteur vis-à-vis du marchand, ce que ne permettrait pas l'utilisation d'EAP-TLS. Cette adaptation est représentée en figure 3.7 pour le cas où la demande de connexion réussit.

Le protocole EAP-TTLS suppose la présence d'un serveur TTLS (Tunneled Transport Layer Security), qui permet de créer un canal sécurisé, d'un serveur AAA (Authentication Authorization Accounting), qui centralise les données d'authentification et autorise l'accès au réseau et d'un point d'accès, qui fait le lien entre le demandeur de connexion et la plateforme AAA. Ici, nous considérons que le terminal du marchand est le point d'accès et que la plateforme de paiement contient le serveur AAA et le serveur TTLS. Pour plus de facilité ici, nous considérons le protocole Radius mais les principes présentés ici sont adaptables à tout protocole AAA. L'environnement d'exécution sécurisée dans le terminal marchand prend en charge les différents messages présentés ici. En particulier, nous supposons que le niveau de sécurité de l'environnement d'exécution sécurisée suffit pour stocker les clés liées au protocole AAA et pour gérer l'accès des porteurs au réseau. Ses capacités sont plus importantes que celles de la SIM marchand, ce qui permet de ne pas trop alourdir le protocole. De plus, l'environnement d'exécution sécurisée gère directement et de manière sécurisée les périphériques dont ceux permettant l'accès au réseau.

Les messages entre l'environnement d'exécution sécurisée du terminal marchand et la SIM ou entre la plateforme de paiement et la SIM sont des requêtes et réponses EAP [IETc]. Les en-têtes HDR correspondent ici aux en-têtes liées à ce format. Les différents messages EAP sont encapsulés dans le protocole ISO 7816 [ISO] entre la SIM du porteur et l'environnement d'exécution sécurisée du porteur.Ils sont ensuite ensuite encapsulés dans un protocole de transmission de proximité entre les terminaux client et marchand. Finalement, ils sont encapsulés dans le protocole Radius entre l'environnement d'exécution sécurisée du terminal marchand et la plateforme. Ce sont les environnements d'exécution sécurisée du porteur et du marchand qui prennent les différentes traductions en charge. Conformément au protocole Radius [IETb], le TEE marchand et la plateforme de paiement utilisent un secret pré-partagé pour, en particulier, chiffrer le message qui informe le TEE marchand du refus ou de l'acceptation par la plateforme de partager la connexion.

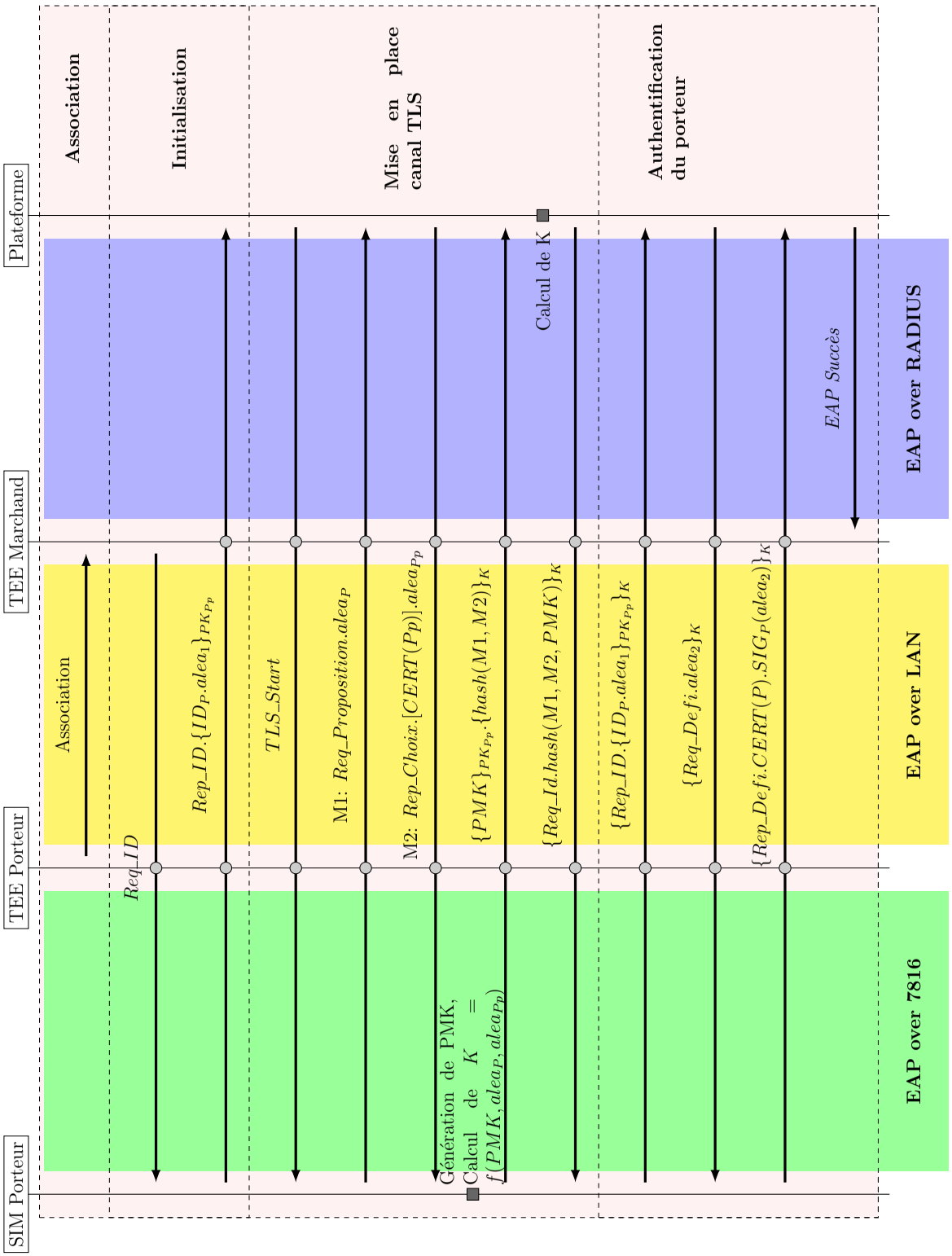


FIGURE 3.7 – Protocole pour le mode semi-connecté

Ce protocole se découpe en quatre phases. Tout d'abord, les deux terminaux sont associés. La deuxième étape permet d'initier la demande de connexion. La SIM porteur transmet l'identité du porteur à la plateforme. Afin de préserver l'anonymat du porteur vis-à-vis du marchand, nous transmettons l'identité du porteur, un aléa la signature de ces deux éléments par la SIM porteur chiffrés avec la clé publique de la plateforme de paiement.

Ensuite, un tunnel TLS est mis en place. Comme le montre la figure 3.7, les messages M1 et M2 correspondent à la proposition et au choix des fonctionnalités cryptographiques. L'envoi du certificat de la plateforme est optionnel. La carte SIM génère la clé pré-maître PMK et calcule la clé secrète K à partir de PMK et des aléas $alea_{Pp}$, $alea_P$ échangés dans M1 et M2. La clé pré-maître chiffrée par la clé secrète de la plateforme de paiement lui est envoyée ainsi que le haché de M1 et M2 chiffré par K. La plateforme doit alors retrouver PMK et recalculer K pour initier la phase d'authentification de la plateforme. Le haché de M1, M2 et PMK permet à la plateforme de prouver à la carte SIM qu'elle possède bien la clé privée PK_{Pp} et qu'elle est capable de calculer K. La clé K est utilisée pour sécuriser les prochains messages.

La réponse d'identité du porteur est renvoyée au serveur qui transmet un défi à la SIM porteur. Celle-ci s'authentifie auprès du serveur en transmettant son certificat et en signant le défi. Si cette authentification réussit, la plateforme autorise le terminal marchand à partager sa connexion avec le terminal client. Cet échange suit le format défini dans [IETf]. En cas de succès, il s'agit d'un message chiffré avec la clé secrète S_{AN} de type EAP-Success et encapsulé dans un message RADIUS Access-Accept.

3.1.6 Transactions en mode tout-déconnecté

Dans cette partie, nous considérons que le transfert se réalise de la même manière qu'il s'agisse d'un paiement marchand ou d'un transfert entre particuliers.

En mode déconnecté, les comptes des porteurs et la plateforme de paiement ne sont plus accessibles. Le mécanisme d'autorisation de la transaction ne peut donc être conservé. Deux solutions existent. La première consiste à associer un porte-monnaie électronique associé au compte en ligne et stocké dans l'appareil. Il serait approvisionné à la demande du porteur à partir de son compte en ligne. Une telle solution a été spécifiée et validée lors d'une collaboration avec le laboratoire Heudiasyc [HB12a, HB12b]. La deuxième solution, que nous adoptons ici, consiste à

autoriser le porteur à effectuer des dépenses hors ligne. Celles-ci seront débitées de son compte lorsque l'un des deux acteurs de la transaction présente à la plateforme des preuves des paiements hors ligne. La plateforme de paiement doit alors déléguer l'autorisation de la transaction aux appareils des acteurs de la transaction. L'intérêt de cette deuxième solution pour l'opérateur est qu'il n'a pas à gérer des problématiques spécifiques au porte-monnaie électronique comme l'atomicité de la monnaie pour en empêcher la création ou la destruction lors d'un transfert entre deux mobiles [Tyg96] ou la disparition de monnaie électronique suite à la destruction ou la perte du support où elle est stockée.

Chaque utilisateur, marchand ou porteur qui souscrit aux services en mode déconnecté, dispose d'un compte à partir duquel les transactions en ligne peuvent être réalisées et un plafond, inscrit dans la carte SIM, qui indique la somme maximale qu'il peut dépenser hors ligne. Seules les établissements ayant un statut de banque peuvent accorder des crédits. Les utilisateurs du service de transactions sur terminaux mobiles n'ont donc pas le droit d'être débiteurs, le plafond ne peut donc dépasser le solde du compte de l'utilisateur. De plus, le protocole de paiement proposé doit garantir l'impossibilité pour un utilisateur de dépenser une somme supérieure à ce qu'il possède. Il est également possible de séparer l'argent pouvant être dépensé hors ligne ou en ligne en associant à chaque utilisateur un compte spécifique à chacun des usages. Nous ne spécifions pas ce cas dans le cadre de cette thèse, cependant, tous les mécanismes définis ci-dessous peuvent être utilisés pour gérer le compte pour les dépenses hors ligne. Les transactions pour transférer de l'argent d'un compte à l'autre peuvent suivre le protocole de transfert entre particuliers que nous proposons ici.

Flux

Les transferts en mode déconnecté ont lieu en deux étapes, une phase de paiement où l'expéditeur et le destinataire valident un contrat de paiement, sans aucune intervention de la plateforme, suivie d'une phase de compensation où la plateforme réalise ce contrat. Lors de la phase de paiement, représentée en figure 3.8, l'expéditeur et le destinataire valident chacun de leur côté la transaction. Lorsque la plateforme traite une transaction pour réaliser la compensation, le compte de l'expéditeur est débité et celui du destinataire est crédité. Comme la plateforme effectue uniquement les mouvements d'argent entre les différents comptes, la validation ne reflète pas seulement l'accord des personnes. Elle reflète également l'autorisation de la plateforme

qui est déléguée aux appareils qui permettent à l'expéditeur et au destinataire de réaliser la transaction. Les autorisations ainsi obtenues sont conservées et transmises ultérieurement à la plateforme pour qu'elle débite et crédite les comptes correspondants. Cela nécessite donc une phase de collecte pour que la plateforme soit informée des différents paiements réalisés, une phase d'actualisation du plafond inscrit dans l'appareil et une phase de mise à jour des règles d'autorisation dans l'appareil de l'expéditeur et du destinataire.

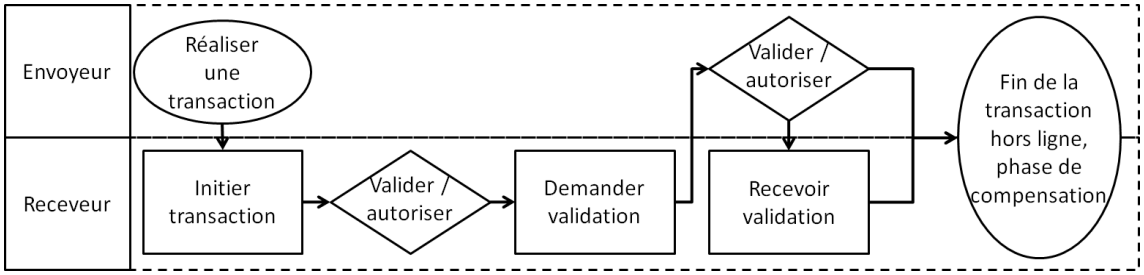


FIGURE 3.8 – Flux de la phase de paiement d'un transfert en mode déconnecté

Pour conserver la traçabilité des transactions, nous proposons que la collecte et les différentes mises à jour s'effectuent de manière forcée à chaque fois que l'un des deux partenaires de transaction réalise une transaction en mode tout-connecté. Pour cela, nous avons modifié les flux des transactions vues précédemment, paiement marchand et transfert entre particuliers, afin d'y inclure la collecte. Le premier échange entre la plateforme et un acteur doit être l'envoi des contrats de paiement. Comme la collecte d'une transaction peut se faire soit par l'expéditeur, soit par le destinataire, il faut éviter que la transaction soit prise deux fois en compte. Une fois les contrats de paiement envoyés à la plateforme, le plafond et les règles d'autorisation sont misà jour.

En résumé, les besoins de sécurité de ce cas d'usage sont les suivants :
- confidentialité, authenticité et intégrité des échanges ;
- anonymat des porteurs vis-à-vis des acteurs externes à la transaction ;
- authentification mutuelle des acteurs de la transaction ;
- preuve permettant à la plateforme d'identifier et d'authentifier les acteurs de la transaction ultérieurement ;
- preuve de la validité du paiement pour le destinataire ;
- non-répudiation de la validation et donc de la transaction par l'expéditeur et le destinataire ;
- authenticité et intégrité de l'autorisation de transaction ;

- impossibilité pour un utilisateur de dépenser une somme supérieure à celle présente sur son compte ;
- impossibilité du rejeu de la preuve de paiement dans le cadre du processus de collecte ;
- protection contre le rejeu de paramètres de règles ;
- authenticité et intégrité des paramètres à mettre à jour.

Protocole

Le protocole permettant de réaliser la phase de paiement d'un transfert en mode déconnecté, représenté en figure 3.9 se divise en trois étapes. Tout d'abord, un canal sécurisé est établi entre l'expéditeur et le destinataire de la transaction. Cette étape est une adaptation du canal sécurisé présentée dans la partie 3.1.2. Elle permet d'assurer une authentification mutuelle des deux acteurs de la transaction et la génération d'une clé secrète partagée SK_{ED}. Celle-ci sert à protéger la confidentialité des messages suivants.

Les deuxième et troisième étapes correspondent à l'accord d'un contrat de paiement par les deux entités en présence. Cet accord est composé d'une autorisation par la carte SIM et d'une validation de l'utilisateur par la saisie d'un secret. Après la mise en place du canal sécurisé, la carte SIM de l'expéditeur demande la validation du destinataire en lui envoyant entre autres un aléa, $alea_E$, comme défi. La carte SIM autorise ou refuse la transaction en se basant sur les règles déléguées par la plateforme de paiement. Si le résultat est négatif, le protocole s'arrête, sinon, le code secret du client est demandé. De même, le protocole continue si la saisie du code PIN par le destinataire réussit. Ensuite, la carte SIM du destinataire choisit un identifiant de transaction TID_D qui lui est propre et le concatène avec TID_ED qui permettra à la plateforme d'identifier la transaction de manière unique. Ensuite, la carte utilise les règles pour autoriser la transaction. Les résultats de l'autorisation et du consentement du destinataire, $Auto_D$ et $Verif_D$ sont envoyés à la carte SIM de l'expéditeur ainsi qu'une signature de ces éléments et du défi $alea_E$. Un défi $alea_D$ est également envoyé. Il devra être utilisé par la SIM de l'expéditeur pour éviter le rejeu de sa partie du contrat. Notons qu'à partir de ces messages, la première partie du contrat $TID_{ED}.TD.TT.TA.CERT_D.Auto_D.Verif_D.alea_D$ $.SIG_D(TID_{ED}.TD.TT.TA.CERT_D.Auto_D.Verif_D.alea_D.alea_E)$ est créée.

Afin d'éviter des messages inutiles, nous avons choisi que la réponse de validation

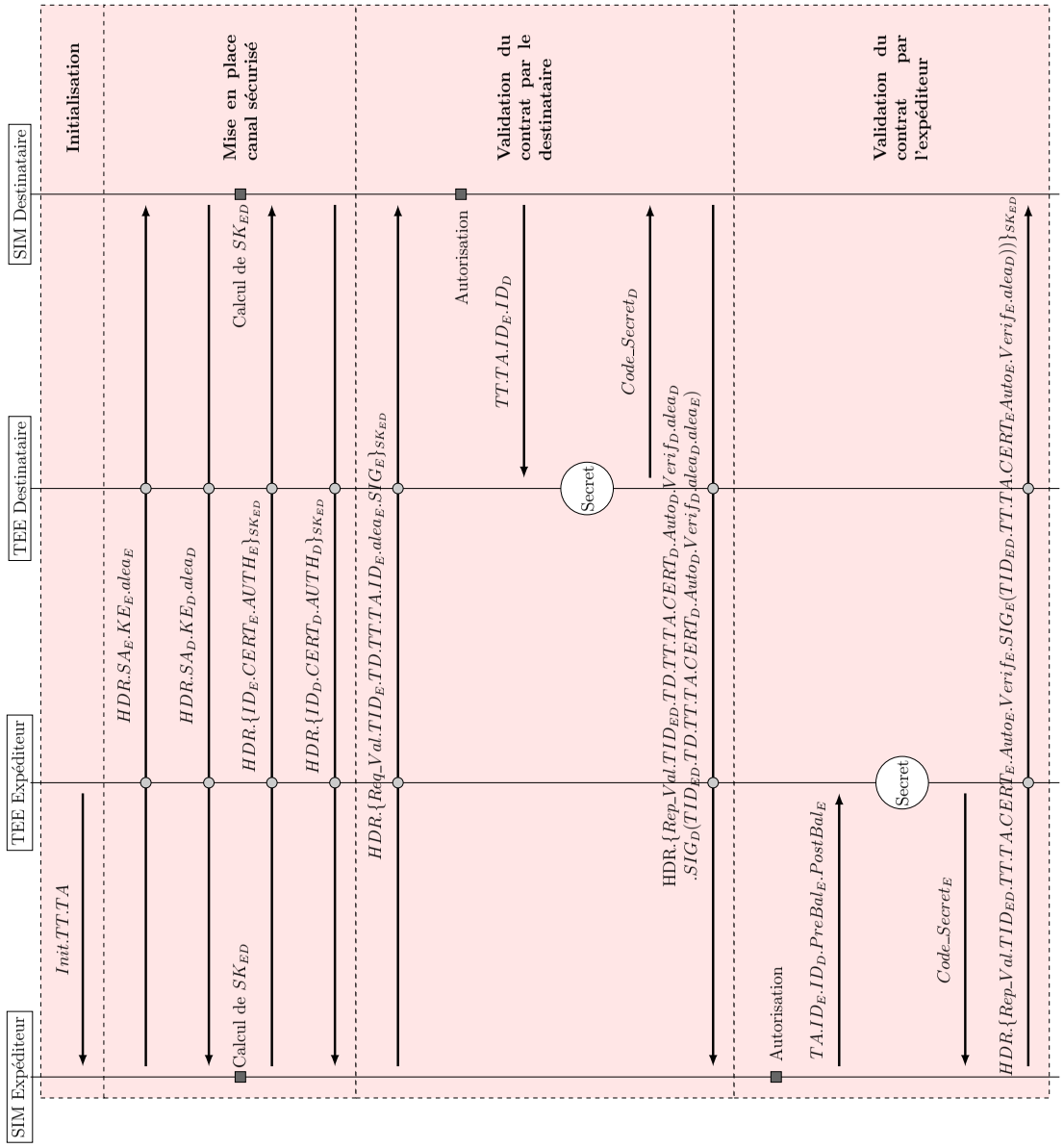


FIGURE 3.9 – Phase de paiement en mode déconnecté

du destinataire serve également à demander la validation de l'expéditeur. De la même manière que précédemment, la transaction est autorisée par la carte SIM et consentie par l'expéditeur. La carte SIM envoie à son tour la réponse de validation constituée des valeurs $Auto_E$, $Verif_E$ ainsi qu'une signature de ces éléments et du défi $alea_D$. Le contenu de ce dernier message $TID_{ED}.TT.TA.CERT_E.Auto_E.Verif_E$ $.SIG_E(TID_{ED}.TT.TA.CERT_E.Auto_E.Verif_E.alea_D$ constitue la seconde partie du contrat de paiement CP_{TID}.

Le contrat de paiement CP_{TID} est détenu par les deux acteurs de la transaction. A la prochaine connexion de l'un d'entre eux, la plateforme de paiement utilisera CP_{TID} pour identifier les acteurs de la transaction et réaliser la compensation du paiement.

Toutes les transactions présentées ici se réalisent en face à face. L'intérêt pour le transfert entre particuliers est que le destinataire reçoit la preuve du paiement au moment même où celui-ci a lieu. Il peut ensuite choisir le moment où il recevra l'argent.

En cas de coupure volontaire ou involontaire de la communication entre les deux acteurs au moment où l'expéditeur envoie sa réponse de validation, le destinataire ne reçoit pas la confirmation du transfert. Par exemple, dans le cas d'un paiement marchand, celui-ci peut alors refuser d'exécuter sa part du contrat et de livrer la marchandise. Si la collecte est ensuite réalisée du côté de l'expéditeur, la plateforme de paiement recevra un contrat CP_{TID} signé par les deux parties et débitera l'expéditeur pour une transaction qui a échoué.

Pour éviter ce cas de figure, nous préconisons que la collecte du côté de l'expéditeur n'ait qu'un rôle informatif pour la plateforme et qu'un mécanisme de reprise soit mis en place. Ce mécanisme est le suivant. Si le destinataire ne reçoit pas la réponse de validation de l'expéditeur au bout d'un certain temps Ta, la carte SIM du destinataire renvoie la requête de validation. Cette opération peut être répétée N fois. Si la validation de l'expéditeur n'a toujours pas été reçue au bout de N tentatives, le contrat est annulé, et le destinataire crée un contrat CP_{TID_echec} qui contient la première partie du contrat ainsi que les différentes requêtes sans réponse. Si la validation de l'expéditeur est reçue, le protocole se déroule normalement. Par cette approche, si la plateforme de paiement reçoit le contrat de paiement par l'expéditeur, elle réserve la somme correspondante sur le compte de l'expéditeur en attendant que le destinataire lui envoie un contrat CP_{TID} conforme à celui de l'expéditeur. Si la plateforme reçoit CP_{TID_echec}, le paiement est annulé et la somme est débloquée.

Afin d'éviter que la somme réservée pour le paiement déconnecté soit bloquée trop longtemps, nous proposons qu'une date limite soit mise en place pour la collecte du côté du destinataire. Au-delà de cette date, la somme est débloquée et le destinataire perd son droit sur le paiement.

Le cas que nous venons d'évoquer montre l'importance du champ TD_X qui

correspond à la date de la transaction. En l'absence de la plateforme au moment du paiement, les appareils de l'expéditeur et du destinataire ont la responsabilité de dater la transaction. Comme celle-ci est un élément de preuve du contrat de paiement et que la carte SIM ne peut générer ce type d'éléments, nous proposons que l'environnement d'exécution sécurisée se charge de fournir la date à la carte SIM pour dater la transaction. Nous supposons que cette date est synchronisée avec la plateforme. La date est d'abord générée par l'expéditeur lors de sa demande de validation, voir figure 3.9. Lorsque la carte SIM du destinataire reçoit la demande de validation, elle compare cette date avec celle fournie par l'environnement d'exécution sécurisée du destinataire. Si la date n'est pas correcte, la transaction ne se poursuit pas.

Deux types de mise à jour des règles d'autorisation existent. La première correspond à l'ajout, la suppression ou toute modification concernant les structures des règles. Nous supposons que ces opérations sont assez rares et peuvent être réalisées lors de la mise à jour de l'application carte. Ce processus n'entre pas dans le cadre de la thèse mais de plus amples informations peuvent être trouvées dans les spécifications de Global Platform [Glo11a]. La deuxième opération concerne la mise à jour des paramètres des règles, le téléparamétrage. Nous proposons que cette opération se réalise en même temps que l'opération de collecte.

La figure 3.10 représente les messages permettant de réaliser la collecte et le téléparamétrage. Ceux-ci précèdent le premier message du protocole de transfert entre particuliers, le deuxième et le cinquième messages du protocole de paiement marchand en mode connecté. L'établissement du canal sécurisé, repris dans la figure 3.10, sert de point de repère pour replacer les messages dans leur contexte.

Après la mise en place de ce canal, la SIM envoie une requête de mise à jour qui comprend les paramètres actuels et le nombre de contrats de paiement stockés dans la carte SIM. Si les paramètres doivent être actualisés et que des contrats de paiement doivent être collectés, la plateforme répond en envoyant Maj, un vecteur des différents paramètres et une requête des contrats. La signature reprend l'aléa du premier message envoyé par la SIM comme garantie contre le rejeu.

La SIM envoie alors les différents contrats de paiement auxquels elle est associée et un acquittement de la mise à jour. La signature reprend l'aléa du message précédent, ce qui permet de prouver la réception des paramètres actualisés. Les différents

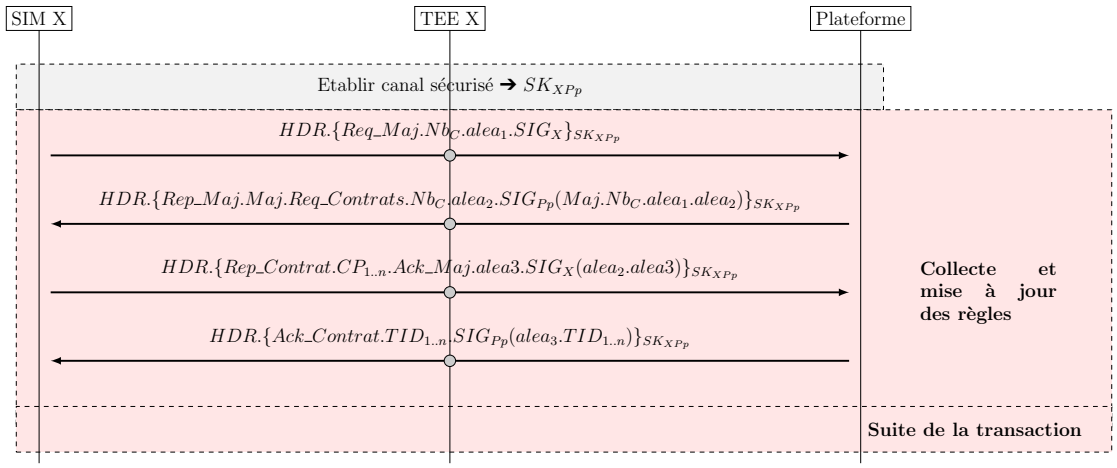


FIGURE 3.10 – Collecte et téléparamétrage

contrats CP_1, CP_2, ... , CP_n réalisés par X n'ont pas besoin d'être signés puisqu'ils contiennent leurs propres protections contre le rejeu. En outre, nous supposons que la plateforme conserve un historique des contrats déjà pris en compte et qu'avant de réaliser une compensation, elle vérifie que le contrat en cours n'est pas déjà présent dans l'historique.

Finalement, lorsque la plateforme a traité tous les contrats collectés, elle envoie un acquittement $TID_{1..n}.SIG_{Pp}(alea_3.TID_{1..n}$ qui correspond à une preuve de la réception du message précédent. La plateforme envoit également l'ensembles identifiants de transaction reçus. A la réception de cette preuve de réception, la liste des contrats de paiement $TID_{1..n}$ est supprimée.

Si les paramètres n'ont pas à être actualisés, le deuxième message du protocole représenté en 3.10 contient une valeur de Maj qui indique à la carte SIM qu'un téléparamétrage n'a pas lieu. S'il n'y a pas de contrat de paiement, la plateforme envoie un acquittement de contrats à la place de la requête dans le deuxième message du protocole représenté en figure 3.10. Dans ce cas, la SIM répond uniquement avec l'acquittement des mises à jour dans le troisième message.

Lorsqu'un utilisateur, porteur ou marchand, réalise une transaction en ligne, le plafond des dépenses autorisé hors ligne peut être amené à évoluer. Pour prendre cela en compte, nous ajoutons quelques traitements à la SIM lors de la phase de confirmation de la transaction des protocoles de transferts entre particuliers. Pour l'expéditeur de la transaction en ligne, lorsque la SIM envoie la validation de la

transaction, elle soustrait cette somme au plafond. La confirmation de la transaction par la plateforme de paiement comprend le nouveau solde du compte et permet ainsi à la SIM soit de confirmer le nouveau plafond soit de le remonter. Quant au destinataire de la transaction en ligne, la validation de la transaction n'entraîne pas de modification du plafond par la SIM mais le nouveau solde envoyé par la plateforme dans la confirmation de la transaction est pris en compte pour calculer le nouveau plafond.

3.2 Validation

Cette section présente comment les protocoles proposés ci-dessus sont validés ainsi que les résultats de cette validation. Celle-ci se déroule en deux étapes. Tout d'abord, les protocoles sont vérifiés formellement afin de montrer qu'ils répondent aux propriétés de sécurité spécifiées. Ensuite, les performances des protocoles sont étudiées et discutées. Cette analyse est nécessaire puisque l'architecture proposée se base sur des éléments disposant de ressources réduites, les cartes à puce.

3.2.1 Vérification formelle

Méthodologie

Différentes méthodes de vérification formelle existent [Col10]. Nous avons choisi de nous baser sur les méthodes automatiques [Col10] qui sont couramment utilisées dans le domaine de la sécurité [HAGTR09]. Parmi les différents outils de l'état de l'art comparés par [CLN09, LTV10, PBP+10], nous avons choisi d'utiliser l'outil AVISPA, Automated Validation of Internet Security Protocoles and Applications, ainsi qu'un module complémentaire SPAN, Security Protocol ANimator for AVISPA. En effet, AVISPA comprend quatre modules d'analyse qui implémentent différents algorithmes de vérification de modèle et de raisonnement [AVI03b], ce qui permet d'avoir une étude plus complète. Ces modules sont On-th-Fly-Model-Checker *OFMC* [BMV05], Constraint-Logic-based Attack Searcher *CL-AtSe* [Tur06], SAT-based Model-Checker *SATMC* [AC04], et Tree Automata based on Automatic Approximations for the Analysis of Security Protocols *TA4SP* [Vig06]. De plus, SPAN facilite la modélisation et l'interprétation des résultats puisqu'il permet de visualiser les protocoles modélisés et les protocoles des attaques trouvées par les différents modules. Finalement, AVISPA permet de modéliser un grand nombre de propriétés de sécurité à partir de deux

propriétés de base, secret et authentification.

Ces propriétés complexes sont spécifiées dans le livrable D6.1 du projet AVISPA [AVI03a]. Le livrable D6.2 de ce même projet montre comment ces propriétés sont modélisées avec AVISPA pour différents protocoles connus [AVI03c]. Les protocoles sont modélisés à l'aide du langage High Level Protocol Specification Language, *HLPSL* [AVI06]. Ce langage modélise un protocole sous la forme d'un graphe d'états. Il permet de spécifier des rôles, les différents messages échangés ainsi que les propriétés de sécurité à respecter. Différents *environnements* sont créés. Ceux-ci permettent d'implémenter des acteurs suivant un rôle spécifique et de les faire interagir dans une session spécifique.

HLPSL permet également de déclarer la connaissance de l'intrus dans l'environnement. AVISPA se base sur le modèle d'intrus Dolev-Yao [Tur03]. Cela revient à considérer que le réseau qui relie les différents acteurs modélisés est corrompu. Dans ce cas de figure, l'intrus contrôle complètement le lien entre les entités impliquées. Il peut intercepter et analyser tous les messages. Si sa connaissance contient les bonnes clés, il est également capable de modifier ou de créer des messages et de se faire passer pour un autre acteur. Il peut décomposer et reconstruire des messages. AVISPA se base sur l'hypothèse que la cryptographie est parfaite. L'intrus ne peut donc pas casser un algorithme particulier. Ce modèle d'intrus correspond bien à notre scénario puisque nous ne faisons confiance ni aux réseaux utilisés ni aux terminaux mobiles.

Afin de vérifier formellement les protocoles proposés, ceux-ci et leurs propriétés de sécurité sont tout d'abord modélisés en HLPSL. L'exactitude de la modélisation est ensuite vérifiée à l'aide de SPAN. Cette étape a pour objectif de vérifier que le protocole modélisé est exact sur le plan de la syntaxe HLPSL, qu'il s'exécute jusqu'à la fin et qu'il correspond bien à la spécification dérite ci-dessus. SPAN permet de visualiser les différents états et messages échangés par les acteurs. Le protocole peut être déroulé et à chaque étape, SPAN propose les transitions qui peuvent être actionnées. On vérifie ainsi que tous les états sont atteignables. Cette étape est nécessaire car un protocole mal modélisé peut s'exécuter sans erreur et sera forcément analysé comme étant sûr.

Les différents modules analysent ensuite les protocoles selon un plan de test spécifiant les différentes sessions à tester. Des sessions uniques et des sessions parallèles ont été utilisées. Nous avons également considéré des cas où l'un des acteurs du

protocole est malveillant. Les résultats de ces deux étapes sont décrits et discutés pour chaque protocole.

TA4SP n'a pas été utilisé pour l'analyse des protocoles de paiement car il ne prend pas en charge la propriété d'authentification. Les modules TA4SP et SATMC n'ont pas été utilisés pour l'analyse de l'établissement du canal de sécurité car ils ne prennent pas en charge l'opérateur exponentiel.

Dans les protocoles où l'environnement d'exécution sécurisée ne joue pas un rôle actif, ils ne font qu'encapsuler ou désencapsuler des messages, nous avons choisi de faire abstraction de ces composants dans la modélisation.

Modélisation des protocoles et plan de test

La spécification HLPSL du protocole permettant d'établir un canal sécurisé a été réalisée à partir de l'exemple *IKEv2-DSx* proposé dans la documentation d'AVISPA [AVI03c]. Nous l'avons complétée en ajoutant des propriétés de sécurité spécifiques. D'autres sessions que celles de l'exemple sont aussi considérées.Nous avons fait abstraction du choix de protocole, le porteur propose un protocole que la plateforme accepte. Les propriétés de sécurité que nous vérifions sont :

- secret vis-à-vis d'un tiers de la clé partagée secrète SK calculée à la suite du Diffie-Hellmann. Cette propriété permet de vérifier que seules l'entité X et la plateforme détiennent la clé secrète ;
- authentification de la clé SK calculée par l'entité X. Cette propriété permet de vérifier que la clé SK provient bien de X ;
- authentification de la clé SK calculée par la plateforme. Cette propriété permet de vérifier que la clé SK provient bien de la plateforme ;
- secret de l'identité de X vis-à-vis d'un tiers extérieur à l'échange. Cette propriété permet de vérifier que ce protocole respecte bien l'anonymat de X ;
- authentification de la clé SK utilisée pour chiffrer le premier message du canal sécurisé envoyé par l'entité X à la plateforme de paiement. Cette propriété permet de vérifier l'authenticité, et par extension l'intégrité, de tous les messages reçus par la plateforme et émis par X ;
- authentification de la clé SK utilisée pour chiffrer le premier message du canal sécurisé envoyé par la plateforme de paiement à l'entité X. Cette propriété permet de vérifier l'authenticité, et par extension l'intégrité, de tous les messages reçus par l'entité X.

Comme dans l'exemple de spécification IKEv2-DSx, l'intrus connaît le paramètre g et la fonction f qui permet de calculer la clé partagée SK, les clés publiques des deux entités X et la plateforme, et les messages qui suivent l'établissement du canal sécurisé. Nous supposons également que l'intrus est un souscrivant malicieux du service de paiement et, qu'à ce titre, il dispose de sa propre paire de clés privée et publique. Il pourrait donc dialoguer avec la plateforme de paiement. Ce protocole a été testé avec une session unique et des sessions parallèles. Le scénario 1T1 correspond à une session unique où l'entité X et la plateforme sont légitimes. Le scénario 1T2 correspond à deux sessions en parallèle où tous les acteurs sont légitimes. Le scénario 1T3 correspond à deux sessions parallèles. La première session se déroule entre la plateforme de paiement, toujours saine, et un porteur sain. La deuxième fait intervenir la plateforme de paiement et un porteur malveillant.

Pour faciliter la modélisation et la lisibilité du protocole modélisé, nous avons supprimé quelques champs jugés non nécessaires à la vérification formelle. Par exemple, les en-têtes des différents messages n'ont pas été conservées non plus. Les champs correspondant au type de transaction, solde précédant et solde suivant la transaction ont été supprimés. Ceux-ci correspondent à des détails de transaction et sont accolés au montant dans les différents messages. Le champ montant, TA a été conservé dans la modélisation et les propriétés de sécurité qui s'appliquent à ce champ s'appliquent à l'ensemble des détails de la transaction. La modélisation de la négociation des algorithmes cryptographiques à utiliser durant le protocole a également été simplifiée. Les propriétés de sécurité vérifiées dans ce protocole sont :

- secret de l'identité du destinataire vis-à-vis d'un tiers externe à la transaction, ce qui permet d'assurer le besoin de sécurité anonymat des acteurs de la transaction ;
- secret de l'identité de l'expéditeur vis-à-vis d'un tiers externe à la transaction, ce qui permet d'assurer le besoin de sécurité anonymat des acteurs de la transaction ;
- secret des détails de la transaction vis-à-vis d'un tiers externe à la transaction. Cette propriété concerne la confidentialité des détails de la transaction ;
- authentification de la validation de la transaction par l'entité X. Cette propriété permet de vérifier si l'origine de la validation du paiement est non-répudiable et non rejouable ;
- authentification de la confirmation de la transaction par la plateforme de paiement. Cette propriété permet de vérifier si l'origine de la notification de la

transaction est non répudiable et non rejouable.

L'intrus connaît la fonction de hachage utilisée, les clés publiques de la plateforme et de l'entité X. Il dispose également de son propre jeu de clés permettant de dialoguer avec la plateforme. Les mêmes sessions que pour l'établissement du canal sécurisé sont considérées. Le scénario 2T1 correspond à une session unique composée uniquement d'acteurs légitimes. Les scénarios 2T2 et 2T3 mettent en œuvre des sessions parallèles. Dans 2T2, tous les acteurs sont légitimes alors que 2T3 fait intervenir une plateforme de paiement saine, un porteur expéditeur sain et un autre malveillant.

Le cookie a été modélisé par $Cookie.SIG_{Pp}(Cookie)$. Ce choix de modélisation nous permet de représenter les caractéristiques du cookie définies en 3.1.4 et correspond à la proposition de cookie que nous avons faite. Dans la modélisation HLPSL, nous ne spécifions pas le contenu du cookie. Les propriétés de sécurité vérifiées dans ce protocole sont :

- secret des détails de la transaction vis-à-vis d'un tiers externe à la transaction. Cette propriété concerne la confidentialité des détails de la transaction ;
- secret de l'identité du porteur vis-à-vis du marchand et d'un tiers externe à la transaction. Cette propriété assure l'anonymat du porteur ;
- authentification du cookie. Cette propriété permet de s'assurer qu'un cookie provient bien de la plateforme de paiement et qu'il n'est pas rejouable ;
- authentification de la validation du porteur pour s'assurer que celle-ci est non-répudiable et non rejouable ;
- authentification de la validation du marchand pour s'assurer que celle-ci est non-répudiable et non rejouable ;
- authentification de la notification envoyée au marchand pour s'assurer que celle-ci est non-répudiable par la plateforme de paiement et non rejouable ;
- authentification de la notification envoyée au porteur pour s'assurer que celle-ci est non-répudiable par la plateforme de paiement et non rejouable.

L'intrus connaît la fonction de hachage, les clés publiques des acteurs, l'identité de plateforme et celle du marchand. Nous supposons également qu'il dispose de son propre jeu de clés lui permettant d'initier un paiement auprès de la plateforme et qu'il peut instancier un canal sécurisé avec la plateforme sécurisée.

Pour tester ce protocole, trois scénarios avec des sessions uniques et six scénarios avec des sessions parallèles ont été créés. Les scénarios 3T1, 3T2, et 3T3 correspondent

aux sessions uniques. 3T1 fait intervenir trois acteurs sains. Le scénario 3T2 met en œuvre un porteur malicieux et un marchand honnête. Le scénario 3T3 met au contraire en oeuvre un porteur honnête et un marchand malicieux. Les scénarios 3T4, 3T7 et 3T9 correspondent respectivement à deux sessions 3T1, aux deux sessions 3T2 et aux deux sessions 3T3 en parallèle. Le scénario 3T5 met en œuvre 3T1 et 3T2 en parallèle et le scénario 3T6 correspond à 3T1 et 3T3 en parallèle. Le scénario 3T8 correspond à 3T2 et 3T3 en parallèle.

La modélisation du protocole TLS fournie avec la documentation d'AVISPA [AVI03c] a été utilisée comme base. Cette modélisation diffère du protocole illustré en 3.7 puisque deux clés partagées sont calculées à partir de PMK et de K. L'une est propre à la plateforme de paiement et l'autre est propre à la SIM du porteur. Nous avons fait abstraction du rôle du point d'accès, l'environnement sécurisé du marchand et des échanges liés au protocole Radius entre le point d'accès et le serveur Radius. D'après le standard RFC 2865 [IETb] qui concerne le service RADIUS, cet échange est chiffré et authentifié par l'utilisation d'un secret partagé qui ne transite pas sur le réseau. Nous avons choisi de modéliser cela par l'utilisation d'une clé secrète partagée. Les propriétés de sécurité vérifiées dans ce protocole sont :

- secret de la clé partagée attribuée au porteur par le protocole TLS ;
- secret de la clé partagée attribuée à la plateforme de paiement par le protocole TLS ;
- authentification de la plateforme de paiement par la carte SIM du porteur ;
- authentification du porteur par la plateforme ;
- authenticité du message autorisant ou interdisant l'environnement sécurisé du marchand de partager sa connexion ;
- secret de l'identité du porteur, ce qui correspond à l'anonymat du porteur vis-à-vis du marchand.

L'intrus connaît toutes les clés publiques, toutes les fonctions de hachage ou de génération des clés secrètes. Pour tester ce protocole, deux scénarios de session unique et deux scénarios avec des sessions parallèles ont été créés. Les scénarios 4T1 et 4T2 correspondent aux sessions. Dans le scénario 4T1, tous les acteurs sont légitimes alors que dans 4T2 le porteur est illégitime. Les scénarios 4T3 et 4T4 correspondent à respectivement deux sessions de type 4T1 en parallèle et deux sessions de type 4T1 et 4T2 en parallèle.

Les propriétés de sécurité vérifiées dans ce protocole sont :

- secret des détails de la transaction ;
- secret de l'identité de l'expéditeur ;
- secret de l'identité du destinataire ;
- authenticité et non rejouabilité de la validation de la transaction par l'expéditeur ;
- authenticité et non rejouabilité de la validation de la transaction par le destinataire.

L'intrus connaît la fonction de hachage et les clés publiques de l'expéditeur, du destinataire et de la plateforme de paiement. Il possède également une paire de clés privée publique liée au système de paiement. Il est donc aussi capable d'établir une session sécurisée avec un destinataire ou un expéditeur. Pour tester ce protocole, sept scénarios ont été créés dont 3 sessions uniques et 4 sessions parallèles. Les sessions uniques 5T1, 5T2 et 5T3 correspondent respectivement à tous les acteurs sont légitimes, tous les acteurs sauf l'expéditeur sont légitimes, tous les acteurs sauf le destinataire est légitime. Le scénario 5T4 correspond à deux sessions 5T1 en parallèle. Le scénario 5T5 correspond à 5T1 et 5T2 en parallèle. La session 5T6 correspond à 5T1 et 5T3 en parallèle. Finalement, 5T7 correspond à 5T2 et 5T3 en parallèle.

Les propriétés de sécurité vérifiées dans ce protocole sont :
- secret des contrats ;
- secret des règles ;
- authenticité de l'ensemble des messages ;
- preuve de la réception des paramètres de règles et des contrats.

L'intrus connaît la fonction de hachage, les clés publiques des acteurs. Il dispose également de sa propre paire de clés publique et privée. Pour tester ce protocole, une session unique et deux sessions parallèles ont été créées. La session unique 6T1 implémente une entité X et un serveur honnête. Le scénario 6T2 correspond à deux sessions 6T1 en parallèle. Le scénario 6T3 correspond à une session 6T1 en parallèle avec une session où l'entité X est malveillante.

Exécutabilité des protocoles modélisés

Une fois modélisés, tous les protocoles ont été visualisés et testés à l'aide de SPAN. Quelques erreurs syntaxiques ont pu être relevées et corrigées. Nous avons également vérifié que tous les protocoles sont exécutables et que leurs états sont tous atteignables.

Résultats des vérifications formelles

Les différents résultats des protocoles ont été regroupés dans le tableau 3.2. Comme nous l'avons évoqué dans la section 3.2.1 concernant la méthodologie de la validation, le module SATMC ne prend pas en charge l'opérateur exponentiel. C'est pour cela que les scénarios 1T1, 1T2 et 1T3 affichent pour ce module le résultat NA, Non Applicable. Le tableau 3.2 montre que l'ensemble des scénarios testés sont sûrs exceptés les scénarios 3T6 et 3T9.

L'attaque trouvée par AVISPA à travers les deux modules OFMC et Cl-Atse est la même pour les deux sessions 3T6 et 3T9.

Cette attaque, représentée en figure 3.11, implique que deux clients réalisent une transaction avec un marchand malveillant. L'attaque se déroule comme suit. Tout d'abord, le marchand/intrus initie une transaction auprès de la plateforme. Le serveur génère ensuite un cookie qu'il envoie au marchand/intrus. Celui-ci envoie le cookie à un client qui répond avec son identité et le cookie chiffrés avec la clé de session négociée entre le client et la plateforme de paiement. L'intrus intercepte cette communication. Il envoie alors le même cookie à un autre client et obtient son identité chiffrée avec sa propre clé secrète négociée avec la plateforme de paiement.

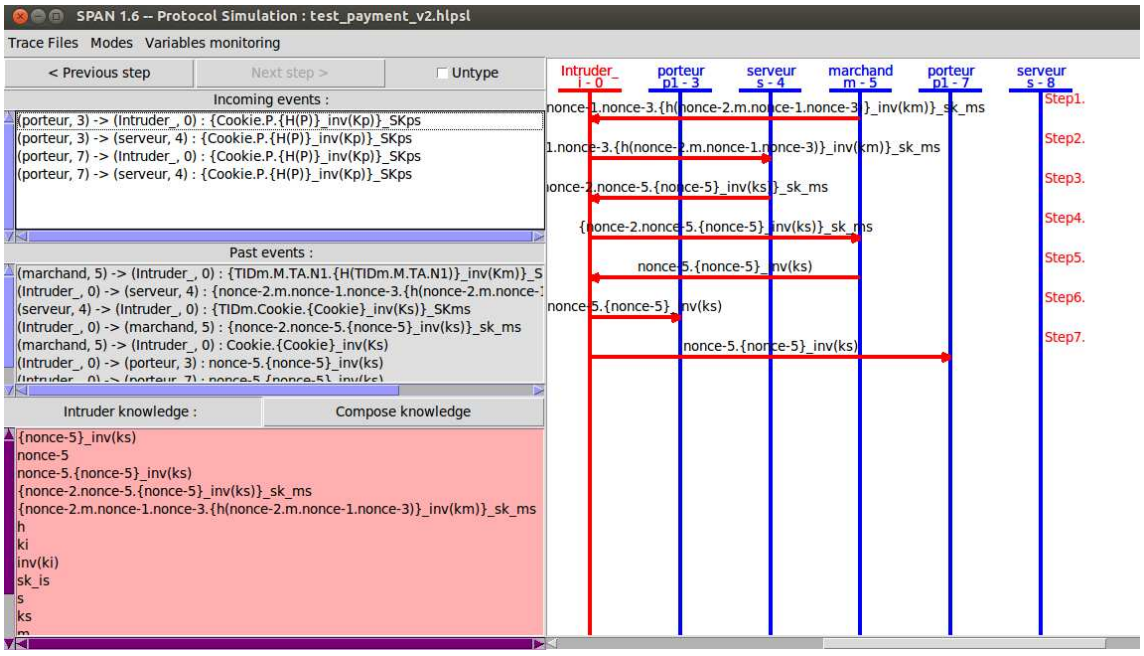


FIGURE 3.11 – Attaque résultant de l'analyse du scénario 3T6 par AVISPA, visualisation par SPAN

TABLE 3.2: Résultats de la vérification formelle des différents protocoles avec AVISPA, modules OFMC, Cl-Atse et SATMC

Protocole	Scénario	OFMC	Cl-Atse	SATMC
Canal sécurisé	1T1	SÛR	SÛR	NA
	1T2	SÛR	SÛR	NA
	1T3	SÛR	SÛR	NA
Transfert entre particuliers	2T1	SÛR	SÛR	SÛR
	2T2	SÛR	SÛR	SÛR
	2T3	SÛR	SÛR	SÛR
Paiement marchand en mode connecté	3T1	SÛR	SÛR	SÛR
	3T2	SÛR	SÛR	SÛR
	3T3	SÛR	SÛR	SÛR
	3T4	SÛR	SÛR	SÛR
	3T5	SÛR	SÛR	SÛR
	3T6	NON SÛR	NON SÛR	SÛR
	3T7	SÛR	SÛR	SÛR
	3T8	SÛR	SÛR	SÛR
	3T9	NON SÛR	NON SÛR	SÛR
Partage de connexion pour paiement semi-connecté	4T1	SÛR	SÛR	SÛR
	4T2	SÛR	SÛR	SÛR
	4T3	SÛR	SÛR	SÛR
	4T3	SÛR	SÛR	SÛR
	4T4	SÛR	SÛR	SÛR
Paiement déconnecté	5T1	SÛR	SÛR	SÛR
	5T2	SÛR	SÛR	SÛR
	5T3	SÛR	SÛR	SÛR
	5T4	SÛR	SÛR	SUR
	5T5	SÛR	SÛR	SÛR
	5T6	SÛR	SÛR	SÛR
	5T7	SÛR	SÛR	SÛR
Collecte et téléparamétrage	6T1	SÛR	SÛR	SÛR [1]
	6T2	SÛR	SÛR	SÛR
	6T3	SÛR	SÛR	SÛR

Le résultat brut fourni par AVISPA peut ne pas paraître intéressant car il semble difficile pour un marchand de piéger deux clients quasiment simultanément lors de leurs passage en caisse et le gain de cette manœuvre nous semble nul. Cependant, en étendant un peu ce scénario, nous concluons que ce protocole est vulnérable à l'attaque *mafia fraud* décrite par Desmedt *et coll.* [DGB06]. En effet, si on considère un porteur A honnête, un restaurateur B membre de la mafia, son complice un autre porteur C et un joaillier honnête D, C peut transférer le cookie de sa transaction avec D à B. B peut alors échanger les cookies et les deux transactions peuvent continuer leur cours. Cela implique que D croit avoir été payé par C alors qu'il est payé par A.

Cette attaque peut avoir lieu en temps réel, le temps de validité du cookie n'est donc pas une contre-mesure assez efficace pour ce cas. Elle est d'autant plus possible que le NFC est vulnérable aux attaques de type homme du milieu [Ali12]. Cependant, comme les marchands s'authentifient auprès de la plateforme de paiement, le point de compromission pourrait être trouvé facilement. De plus, des indications concernant le paiement sont envoyées par la plateforme à la carte SIM du porteur sans intermédiaire. Leur affichage ne peut être corrompu car cette tâche est réalisée par l'environnement d'exécution sécurisé. Les deux acteurs honnêtes ont donc un moyen de repérer que le paiement réalisé n'est pas le bon. Ces méthodes ne permettent pas forcément de bloquer la fraude. Les protocoles délimiteurs de distance, *distance-bounding protocols* peuvent être utilisés pour résoudre cette problématique [AK13].

3.2.2 Etude des performances

Méthodologie

L'objectif de cette analyse est d'évaluer le temps requis pour réaliser les différents protocoles de paiement décrits ci-dessus. Le composant de l'architecture proposée qui présente le plus de contraintes, tant du point de vue de capacité de calculs que celui de vitesse de transmission d'informations, est la carte SIM. Les temps associés aux calculs sur les autres composants, l'environnement d'exécution sécurisée et la plateforme de paiement sont négligés ainsi que les temps de transmission des données sur les interfaces autres que celles de la carte. Les temps considérés ici correspondent donc au temps passé en calculs cryptographiques dans la carte SIM ainsi que le temps de transmission des différents messages de et vers la carte.

L'étude réalisée est une analyse théorique. La durée de différentes opérations

cryptographiques de base, comme un chiffrement ou un hachage, a été mesurée. Les différentes durées obtenues servent ensuite à calculer les temps de calculs nécessaires pour chaque protocole. Les durées de transmission sont quant à elles évaluées à partir du débit théorique d'entrée et de sortie des cartes à puce.

Les différents algorithmes considérés pour cette étude sont regroupés dans la table 3.3. La fonction de hachage choisie est SHA-1. Les chiffrements symétriques sont réalisés avec l'algorithme 3-DES et une clé de 168 bits. L'algorithme de chiffrement asymétrique utilisé est RSA avec une clé de 1024 bits. Pour signer un message, celui-ci est d'abord haché puis signé en suivant l'algorithme RSA avec un clé de 1024 bits.

TABLE 3.3: Algorithmes pris en compte pour l'analyse des performances

Opération cryptographique	Algorithme
Hachage	SHA-1
Chiffrement symétrique	3-DES 168 bits
Chiffrement asymétrique	RSA 1024 bits
Signature	RSA 1024 bits

La charge utile des messages des différents protocoles peut être divisée en trois catégories selon leur taille. Ceux de plus petite taille ont une longueur inférieure à 365 octets. En incluant la signature de la charge utile, la taille du message est inférieure à 500 octets. Les messages de plus grande taille sont ceux qui comprennent un certificat X509. Leur taille estimée est inférieure à 2 kilo-octets. En incluant la signature, la taille du message est inférieure à 2 500 octets. Il existe quelques messages de taille moyenne, environ 1 kilo-octet, mais ils ne sont ni chiffrés ni signés.

Afin de mesurer les temps des différentes opérations cryptographiques, les messages ont été regroupés en deux catégories, taille inférieure à 500 octets et taille inférieure à 2 500 octets. De cette manière, la durée pour réaliser l'opération sur un message d'une catégorie est approximée par la mesure correspondant à la catégorie. Par exemple, un message de taille 300 octets appartient au premier groupe. Son temps de chiffrement sera donc approximé par le temps de chiffrement d'un message de 500 octets.

Le tableau 3.4 regroupe les mesures nécessaires à l'évaluation des performances des différents protocoles. Le temps requis pour hasher, signer, vérifier la signature, chiffrer et déchiffrer symétriquement un message de 500 ou 2 500 octets. Le temps pour chiffrer un message de 16 octets avec RSA est aussi mesuré. Cette opération

n'est réalisée que dans un seul message c'est pour cela que la taille du message a été prise en compte et pas une des deux catégories définies ci-dessus. Différents aléas sont nécessaires dans les protocoles. Nous avons supposé que tous font 260 octets, soit la taille maximale d'un aléa prévu dans le protocole IKEv2. Le protocole TLS, quant à lui, nécessite le choix d'un aléa de taille 48 octets comme clé pré-maître. Ces différentes mesures ont été réalisées sur une carte SIM avec un CPU de 32 bits et une RAM de 32 kilo octets. Ces temps correspondent à la moyenne sur 100 exécutions dans la carte à puce. Le temps moyen d'envoi de l'instruction de calcul est ensuite retranché pour obtenir les mesures présentées dans le tableau 3.4. Le temps de calcul du secret Diffie-Hellmann g^{ab} est issu de [MM05]. Ce temps n'a pas pu être mesuré car la carte utilisée ne supporte pas l'algorithme Diffie-Hellmann bien que celui-ci est prévu dans la norme Javacard. Enfin, le temps de transmission des messages a été calculé pour une vitesse théorique de 9 600 bits par seconde [Eve92].

TABLE 3.4: Durée de différentes opérations cryptographiques

Taille majorée (octets)	Opération	Temps (ms)
128	chiffrement asymétrique	33
	choix aléa 48 octets	2
	choix aléa 260 octets	12
	secret Diffie Hellman [2]	300
500	hash	6
	signature	168
	vérification signature	37
	chiffrement symétrique	125
	déchiffrement symétrique	121
	transmission	425
2 500	hash	26
	signature	188
	vérification signature	57
	chiffrement symétrique	615
	déchiffrement symétrique	622
	transmission	2 125
1 000	transmission	833

Estimation des performances

Afin d'établir un canal sécurisé au niveau applicatif, deux aléas de 260 octets sont choisis, un secret Diffie-Hellmann est calculé, deux signatures de messages de 2,5 kilo-octets sont réalisées ainsi que leur vérification. Au niveau de la carte SIM,

un message de 2,5 kilo-octets est chiffré et un autre de la même taille est déchiffré. Enfin, trois messages de 1 kilo-octets et un message de 2,5 kilo-octets sont échangés. Le temps de traitement par la SIM est de 3 931 millisecondes, soit 3,93 secondes.

La procédure de transfert entre particuliers requiert la signature et sa vérification de deux messages de 500 octets, le chiffrement et le déchiffrement de deux messages de 500 octets, le choix de 3 aléas de 260 octets et la transmission de quatre messages de 500 octets. Le temps de traitement du paiement seul est de 2638 millisecondes soit 2,64 secondes. Le temps total incluant la mise en place d'un canal sécurisé entre la SIM et la plateforme de paiement est donc de 6,57 secondes.

Pour réaliser un paiement marchand, trois signatures et quatre vérifications de signatures de messages de 500 octets sont réalisées. Trois messages de 500 octets sont chiffrés et déchiffrés par les cartes SIM. Un seul aléa de 260 octets est généré par une carte et 8 messages de 500 octets sont échangés. La durée de la procédure de paiement est donc de 4 377 millisecondes, soit 4,38 secondes. Cette procédure nécessite la mise en place de deux canaux sécurisés, l'un entre l'élément sécurisé du marchand et la plateforme de paiement, l'autre entre l'élément sécurisé du porteur et la plateforme. La durée totale de la procédure de paiement est donc de 12,24 secondes.

Le mode semi-connecté nécessite le partage de la connexion du marchand avec le porteur. Cette étape est réalisée à travers le protocole EAP-TTLS. Celui-ci requiert le chiffrement asymétrique d'un message de 128 octets, le hachage de deux messages de taille inférieure à 500 octets, le chiffrement et le déchiffrement de deux messages de taille 500 octets, le chiffrement d'un message de grande taille, le choix d'un aléa de 48 octets et de deux aléas de 260 octets. huit messages de moins de 500 octets sont transmis et un seul de 2,5 kilo-octets. Cela induit un temps de traitement par l'élément sécurisé de 6 736 millisecondes soit 6,74 secondes. La procédure paiement en mode semi-connecté correspond au temps de partage de la connexion, 6,74 secondes, additionné au temps du paiement marchand, 12,24 secondes, soit en tout 18,98 secondes.

Le mode déconnecté nécessite la mise en place d'un canal sécurisé entre les deux éléments sécurisés puis la création et la validation du contrat de paiement par les deux parties. La mise en place du canal sécurisé nécessite le choix de deux aléas de 260 octets, la génération de deux secrets Diffie-Hellmann, la signature et la vérification de deux messages de 2,5 kilo octets ainsi que le chiffrement et déchiffrement symétrique de ces deux mêmes messages. Deux messages de 500 octets et deux messages de 2,5

kilo-octets sont échangés. Le temps de mise en place du canal sécurisé qui en résulte est de 8 688 millisecondes soit 8,69 secondes. La création et la validation du contrat de paiement se réalise avec la signature et sa vérification de 4 messages de 500 octets, le chiffrement de 4 messages de 500 octets, le choix de deux aléas de 260 octets et la transmission de 6 messages de 500 octets. Comme précédemment, les messages échangés ont été comptabilisés deux fois pour prendre en compte la réception et l'envoi par les deux éléments sécurisés. Le temps de traitement qui résulte de cette étape est de 2 611 millisecondes, soit 2,61 secondes. En tout, la procédure de paiement déconnecté dure 11,29 secondes. Durant cette phase, un contrat de paiement de 5 kilo-octets est créé. La transmission d'un contrat à la plateforme de paiement durant la phase de collecte est de 4,17 secondes.

Discussion sur la validation des protocoles

L'étude des performances menée ici est une analyse théorique réalisée à partir de la mesure des temps de calcul pour des tailles de message majorées. Ces mesures ont été réalisées sur une carte à puce qui ne dispose pas d'un cryptoprocesseur particulier. Les temps de calcul présentés ici sont donc certainement supérieurs à ceux qui peuvent être observés sur les meilleures cartes à puce.

Selon [UPK11], le temps raisonnable pour une transaction est d'environ 5 secondes. Les performances des protocoles présentées ici ne respectent pas ce temps. Il est possible d'améliorer ces performances en optimisant les procédures pour paralléliser certaines étapes. Par exemple, dans le protocole de paiement tout-connecté, le canal sécurisé entre le porteur et la plateforme pourrait être mis en place pendant l'initiation de la transaction par le marchand. La durée du protocole semi-connecté pourrait aussi être réduite si la négociation du canal sécurisé entre le marchand et la plateforme de paiement coincide avec le partage de connexion avec le porteur.

Les messages les plus longs des différents protocoles sont ceux qui comprennent un certificat. Dans cette étude, la taille d'un certificat est estimée à 2 kilo-octets. Pour réduire les durées des protocoles, il serait intéressant de limiter la taille de ces certificats.

L'étude considère une vitesse de transmission de 9 600 bits par seconde. Cependant, certaines cartes peuvent transmettre des données à 115 000 bits par seconde. L'utilisation de telles cartes réduiraient drastiquement les temps de transmission. En effet, à cette vitesse-là, la durée de mise en place d'un canal sécurisé est de 1,98

secondes, la durée d'un transfert entre particuliers est de 3,06 secondes, la durée d'un paiement en mode tout-connecté est de 5,65 secondes, la durée d'un paiement en mode semi-connecté est de 7,32 secondes et la durée d'un paiement en mode déconnecté est de 5,98 secondes. Lors de la collecte, la transmission d'un contrat prendrait alors 0,35 secondes

Une question se pose quant à l'ergonomie du paiement et des communications entre les terminaux. Par exemple, lors d'un paiement déconnecté, il est nécessaire d'échanger trois fois des données entre le terminal marchand et le terminal client. Si à chaque fois, il est nécessaire d'approcher les deux terminaux, la procédure de paiement peut devenir rébarbative pour les porteurs comme pour les marchands. Pour résoudre cette problématique, il faudrait que la connexion établie lors du premier rapprochement persiste tant que les deux acteurs restent à une certaine distance.

3.3 Discussion

Dans ce chapitre, nous avons proposé divers protocoles pour réaliser une sécurité de bout en bout entre l'application de paiement sur la carte SIM des utilisateurs et la plateforme de paiement. Différents modes, tout-connecté, semi-connecté et déconnecté, ont été considérés.

Les différents protocoles ont été vérifiés formellement en tenant compte de différents scénarios définis dans un plan de test. Nous avons ainsi découvert qu'une attaque, variante de la *fraude de la mafia* défini par Desmedt *et coll.* [DGB06], est possible. Différents mécanismes que nous proposons, absents du cas d'usage présent par Desmedt *et coll.* [DGB06] comme la communication des informations concernant la transaction directement entre le serveur et l'élément sécurisé ; et l'utilisation d'un périphérique sécurisé permettent de réduire cette menace. D'autres techniques comme les protocoles délimiteurs de distance, *distance-bounding protocols* [AK13] peuvent aussi être utilisés pour résoudre cette problématique. De plus, le marchand malveillant impliqué dans ce type de fraudes serait facilement identifiable.

Les performances des différents protocoles ont été estimés grâce à une carte à puce. Certains temps présentés dans ce manuscrit ne semblent pas acceptables du point de vue de l'usage. Cela peut être dû au fait que la carte à puce utilisée pour les mesures ne présente pas de cryptoprocesseur particulier. De plus, il est possible d'optimiser certains protocoles pour que des opérations soient réalisées en parallèle.

C'est le cas du protocole de paiement semi-connecté, où le partage de la connexion pourrait être réalisé en parallèle de la négociation du canal sécurisé entre la SIM du terminal du marchand et la plateforme de paiement. La taille des certificats pourrait être limitée. Enfin, des cartes ayant des vitesses de transmission plus rapides pourraient également être utilisées.

Avec de telles cartes, les différentes phases de paiement ont une durée de l'ordre de 5 secondes. La collecte serait elle aussi raccourcie puisque le transmission d'un contrat durerait uniquement 350 millisecondes. Ces performances sont acceptables.

Le paiement en mode semi-connecté aurait encore une durée d'environ 7 secondes. Cependant, comme le processus est différent de celui d'un paiement par carte bancaire dans lequel se placent Urien *et coll.* [UPK11], l'application de la règle des 5 secondes pourrait être discutée ici. L'acceptabilité par les porteurs de ce processus de paiement particulier pourrait être étudiée pour voir si ce temps pourrait convenir.

Chapitre 4

Détection de fraudes pour les services de transactions sur terminaux mobiles

Ce chapitre est consacré aux contributions dans le domaine de la détection de fraudes comportementales pour les services de transactions sur terminaux mobiles. Nous présentons un simulateur de transactions que nous avons conçu et développé. Sa conception est basée sur l'exploitation de données réelles. Les données synthétiques ainsi générées ont été utilisées pour adapter des algorithmes de classification aux services de transactions sur terminaux mobiles.

Sommaire

4.1 Génération de données synthétiques

La détection de fraudes permet de déceler, de contourner ou de contenir des fraudes lorsque l'architecture de sécurité ne les a pas bloquées. L'introduction des technologies et des modèles de paiement implique d'ailleurs une évolution parallèle des fraudes et des fraudeurs qui s'adaptent et trouvent de nouvelles manières d'éviter les fonctionnalités de sécurité. Cependant, ce domaine de recherche est limité car les bases de données publiques contenant des cas de fraudes sont rares comme l'indiquent

Bolton et Hand ainsi que Phua *et coll.* [BH01, PLSMG05].

De plus, la comparaison des algorithmes existants n'est pas fiable puisqu'il n'existe pas de base de données de référence et que la vérité terrain est inconnue ou incertaine. Cette situation résulte du fait que ces données sont sensibles pour les acteurs du paiement autant pour ne pas diffuser des modes opératoires et les vulnérabilités des systèmes que pour protéger les données de la clientèle. C'est d'autant plus problématique dans un contexte de recherche scientifique car les études doivent être reproductibles. Utiliser des bases de données communes est une bonne pratique qui permet la reproductibilité et la validation des résultats.

Dans le cas des transactions sur terminaux mobiles, du fait de leur introduction récente et de la phase de développement actuelle, ces difficultés s'ajoutent à un manque de retour d'expérience sur la fraude. En effet, les systèmes de transaction sur mobile ont commencé à se développer depuis une dizaine d'années. Ils ont connu un réel essor avec le lancement et le succès de M-Pesa en 2007. Par contre, les systèmes de transaction sur carte bancaire existent depuis une trentaine d'années.

Cette thèse repose sur l'utilisation d'une base de données provenant d'un système déployé sur le terrain. Cependant, celle-ci est confidentielle et n'est pas exploitable pour la détection de fraude puisqu'elle ne contient aucune vérité terrain. Afin de résoudre ce problème, nous avons décidé de créer des données synthétiques comme le suggèrent Phua *et coll.* [PLSMG05]. L'originalité de ce générateur de données est de reproduire le fonctionnement de la plateforme de paiement ainsi que les comportements des utilisateurs à partir des données réelles à notre disposition.

Dans ce chapitre, les travaux existants sur l'utilisation de données synthétiques et leur création pour la détection de fraude sont recensés et décrits dans un premier temps. Notre modélisation d'un système de transactions sur mobile est ensuite présenté. Celui-ci est spécifié dans le livrable [AGG+11] sur lequel le projet Européen FP7 MASSIF s'appuie. Une première validation de ce modèle a été réalisée et est détaillée dans ce chapitre. Nous décrivons ensuite comment nous avons paramétré ce simulateur à partir des données réelles. Enfin, des jeux de données ainsi générés sont utilisés pour adapter d'algorithmes de classification à notre cas d'usage.

Comme indiqué dans le chapitre 1, les contributions présentées ici sont limitées aux fraudes comportementales.

4.1.1 Étude de l'existant

Utilisation de données synthétiques

Les données synthétiques sont couramment utilisées dans le domaine de la reconnaissance de formes et de l'apprentissage automatique bien qu'elles impliquent quelques inconvénients. En effet, il est possible que les fonctionnalités observées sur les données synthétiques ne se reproduisent pas sur les données réelles. De plus, les données, frauduleuses ou non, synthétiques peuvent être pas assez réalistes [LKJ02].

Malgré ces inconvénients, certains comme Lundin *et coll.* [LKJ02], argumentent que l'utilisation de données synthétiques permet d'éviter des inconvénients liés aux données réelles. Certaines propriétés requises pour optimiser l'étude des algorithmes ne sont pas vérifiées par ces dernières. Par exemple, certains algorithmes nécessitent de grandes quantités de données labellisées avec une sur-représentation d'événements frauduleux. De telles données ne sont pas forcement disponibles dans les systèmes réels [LKJ02]. La quantité de données pour des tests de résistance ou *stress test* n'est pas forcément disponible non plus [LKJ02]. De manière générale, les intérêts majeurs d'un générateur de données synthétiques sont :

- la possibilité de générer autant de données et de scénarios que nécessaire ;
- le contrôle des paramètres des données générées pour par exemple tester des fonctionnalités spécifiques des algorithmes ;
- la présence de labels pour faciliter l'évaluation des performances ;
- le respect de la vie privée des utilisateurs du système et de manière générale la réduction des problématiques liées à la confidentialité des données ;
- la possibilité d'évaluer des algorithmes pour des systèmes qui ne sont pas encore déployés ou sur lesquels peu de retour d'expérience existe.

Compte tenu de ces caractéristiques, nous considérons que ces deux types de données sont complémentaires. Nous estimons qu'une étude complète et fiable des algorithmes de détection de fraudes nécessite d'utiliser des données réelles et synthétiques.

Création de données synthétiques pour la détection de fraudes

Paradoxalement, bien que très peu de données réelles soient disponibles publiquement, les données synthétiques sont peu utilisées dans le domaine de la détection de la fraude [BH01, PLSMG05]. A notre connaissance, seuls deux générateurs de

données artificielles existent dans ce domaine.

Le premier, spécifié par Barse *et coll.* et Lundin *et coll.* [BKJ03, LKJ02] a été développé pour la détection de fraudes dans un système de vidéo à la demande. Celui-ci modélise le système. Le comportement des utilisateurs est également modélisé grâce à un automate paramétré à partir de données réelles.

Outre le fait que nos deux contextes sont très différents, notre générateur peut être paramétré indépendamment des données réelles. Il peut donc créer des parcours clients fraudeurs prédéterminés qui ne sont pas réalistes mais qui peuvent permettre d'évaluer des caractéristiques particulières. Par exemple, il est possible de créer un groupe d'utilisateurs qui changent fréquemment d'habitudes et un autre groupe qui n'en change pas pour observer l'incidence de ce paramètre sur les algorithmes de détection. De plus, Barse *et coll.* et Lundin *et coll.* [BKJ03, LKJ02] n'évaluent pas le modèle et les données qu'ils génèrent. Une telle évaluation a été réalisée pour valider le simulateur proposé ici.

Le second générateur, proposé par Lopez *et coll.* [LRA12], est très proche de notre contexte puisqu'il cible également les systèmes de transactions sur terminaux mobile. Les cas étudiés ne sont pas similaires puisque Lopez *et coll.* se concentrent sur la détection de blanchiment d'argent alors que nous ciblons les fraudes comportementales définies dans [BJTW11]. Ces dernières se traduisent par un changement de comportement dû au fait qu'un compte est utilisé en même temps par un utilisateur légitime et un fraudeur. Notre modèle du comportement des utilisateurs est plus riche que celui proposé par Lopez *et coll.* [LRA12]. Nous ne nous contentons pas de générer des transactions aléatoires à un moment aléatoire, nous utilisons la notion d'habitude pour modéliser les comportements des utilisateurs légitimes sous la forme de motifs. Nous recréons ainsi la complexité logique des comportements des utilisateurs. Dans notre générateur, un utilisateur peut ainsi être modélisé par des habitudes multiples et par des transactions aléatoires.

D'autres méthodes peuvent être utilisées pour créer des données synthétiques. Cependant, elles ne ciblent pas la détection de la fraude. La démarche la plus proche et qui concerne le domaine du paiement est proposée par Jeske *et coll.* [JSL+05]. Ce générateur permet de créer des données relatives à des transactions par carte de crédit. Cependant, il cible l'évaluation des méthodes de fouilles de données en générale et ne prend pas en considération la modélisation d'attaques ou de fraude.

4.1.2 Modèle et implémentation

Le système modélisé est le système de transactions sur mobiles décrit dans la section 1.2 dans le livrable 2.1.1 du projet MASSIF [AGG+11]. Pour rappel, ce service permet à des porteurs de réaliser diverses transactions, achats ou transferts à l'aide de monnaie électronique, m-monnaie, émise par l'opérateur. Cette monnaie peut être acquise ou échangée auprès de distributeurs. Elle correspond à un compte sous la responsabilité de l'opérateur qui propose le service.

La réalisation de ce simulateur est inspirée de la méthodologie proposée par Lundin *et coll.*, [LKJ02] et représentée par la figure 4.1. Comme suggéré dans cette méthodologie, la simulation de la plateforme, du comportement des utilisateurs et des profils d'utilisateurs est gérée par trois modules distincts. L'architecture du générateur de données ainsi obtenu est représentée en figure 4.2. Cette architecture est décrite ci-dessous.

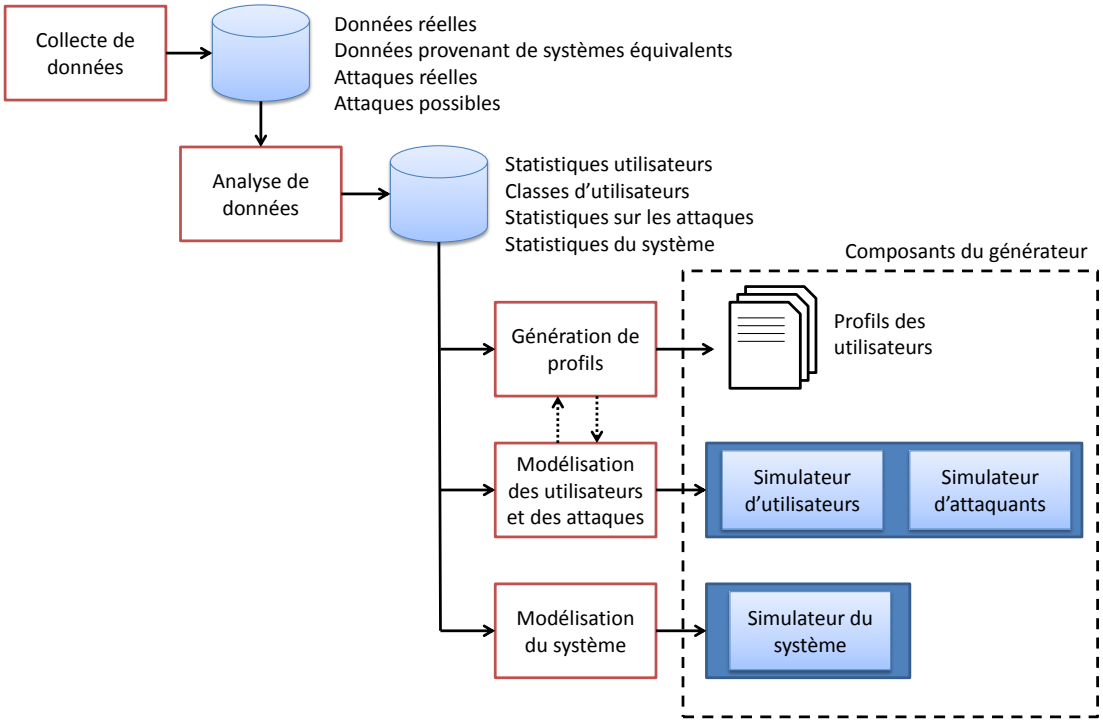


FIGURE 4.1 – Méthodologie de génération de données synthétiques, source [LKJ02]

Modélisation de la plateforme de paiement

Le module de simulation de la plateforme de paiement est constitué d'une interface client, d'un module de gestion des comptes et d'une base de données qui

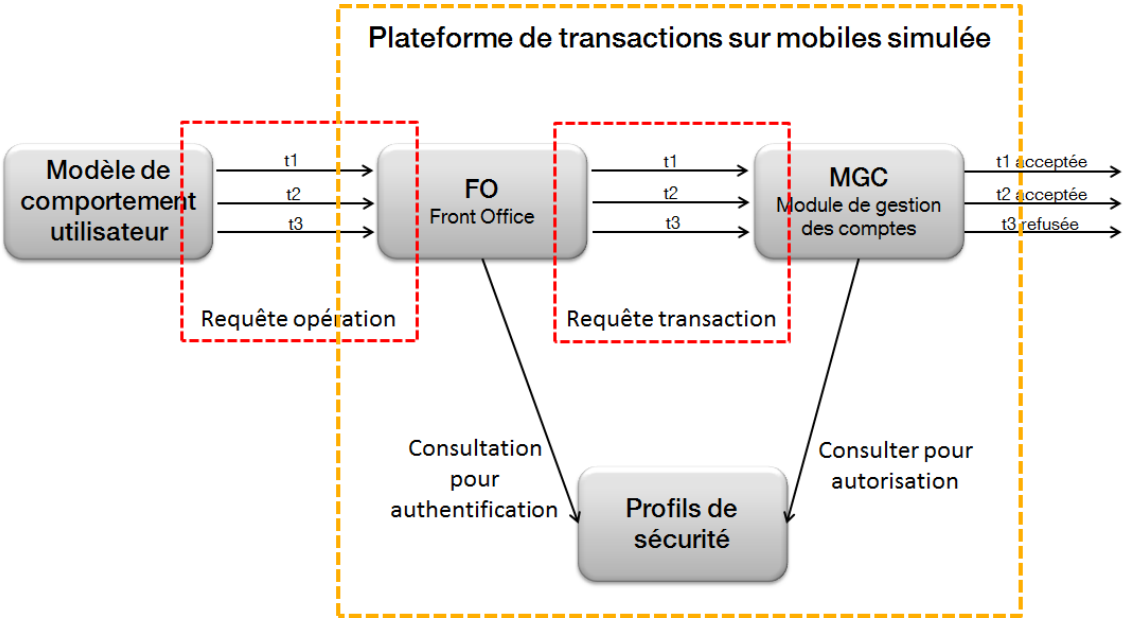


FIGURE 4.2 – Architecture du générateur de données synthétiques

contient les profils des utilisateurs. L'interface client est chargée de filtrer les requêtes d'opération. Celles-ci correspondent aux tentatives de connexion, les demandes de transaction ou des opérations particulières par exemple la modification d'un mot de passe. En particulier, cette interface authentifie les utilisateurs qui souhaitent accéder au service à l'aide d'un mot de passe. Cette fonction particulière a été modélisée. Dans notre simulateur, les mots de passe clients ont été stockés dans les profils des clients. Le module de gestion des comptes réalise les débits ou les crédits relatifs à une transaction. Il autorise également les transactions en fonction du profil des clients.

Un paiement suit la séquence suivante décrite dans [AGG+11] :

1. authentification des acteurs de la transaction ;
2. transmission des instructions de paiement et des détails de la transaction à la plateforme ;
3. autorisation ou refus de la transaction par la plateforme ;
4. crédit et débit des comptes destinataire et expéditeur.

Le générateur enregistre les données liées à l'authentification, échec ou réussite, ainsi que celles concernant l'autorisation d'une transaction.

Modélisation du comportement des utilisateurs

Les instructions de paiement traitées par la plateforme simulée proviennent d'un modèle comportemental des utilisateurs. L'hypothèse, notée A1, sur laquelle se base ce simulateur est que la richesse et la difficulté d'interprétation des journaux de transactions sont la conséquence de l'imbrication d'actions et de comportements logiques réalisés en parallèle par plusieurs acteurs. Nous avons donc choisi une approche de type multi-agents [Fer95] pour, conformément à l'hypothèse A1, modéliser des comportements individuels qui se combinent pour former un ensemble plus complexe. Les agents du système multi-agents sont les acteurs du système, qu'il s'agisse des utilisateurs légitimes qui souscrivent au service de transactions sur mobile ou des fraudeurs qui attaquent le système.

Trois catégories d'acteurs légitimes sont impliqués dans le système de transactions sur terminaux mobiles. Chacune d'elles correspond à un rôle différent et est associée à des actions spécifiques. Comme dans le système réel, les *porteurs* sont des particuliers qui réalisent des transferts ou des paiements à l'aide de leur terminal ; les *marchands* sont des fournisseurs de biens et de services qui acceptent de la monnaie électronique comme règlement des paiements ; les *agents* ou *distributeurs de monnaie électronique* sont des marchands spécifiques qui vendent la monnaie électronique et permettent sa distribution de l'opérateur qui l'émet aux porteurs ou marchands.

Le modèle comportemental des utilisateurs légitimes est basé sur l'hypothèse A2 selon laquelle leurs transactions sont reliées à leurs habitudes. Ceci implique que les utilisateurs légitimes ont tendance à réaliser de manière répétitive et assez fréquente un ensemble spécifique de transactions. Cette hypothèse est aussi prise en compte dans le cadre des méthodes de détection de fraudes basées sur la détection d'anomalies [CBK09, Kok97] où les transactions sont classées comme normales ou anormales. Il est généralement considéré dans ce domaine que les transactions normales correspondent aux habitudes des utilisateurs et que les fraudes diffèrent inévitablement des transactions normales et sont donc un sous-ensemble des transactions anormales.

La modélisation des habitudes des porteurs se base sur la définition de ce concept par Kokkinaki [Kok97] selon laquelle une habitude est une *classe d'équivalence* sur les transactions légitimes. Cette définition est étendue dans le cadre de cette thèse. Une habitude est une répétition d'une séquence de transactions légitimes caractérisées par (1) un type de transaction, (2) un montant qui suit une distribution normale, (3) un écart de temps entre deux transactions qui suit également une distribution normale,

(4) une date initiale et (5) une date de fin. Ces deux dates permettent de définir la durée sur laquelle l'habitude a lieu. Les travaux présentés ici se concentrent sur des habitudes constituées d'une seule transaction et pas une séquence de transactions.

La seconde hypothèse A2 sur laquelle se base ces travaux peut se traduire par : le comportement C d'un porteur est composé d'un ensemble d'habitudes $C = \{H_1, H_2, ..., H_i\}$, où H_i est une habitude pour un type de transaction. L'analyse du système de transactions sur terminaux mobiles par Jack *et coll.* [JTT10] nous conduit à penser que cette hypothèse s'applique également aux transactions réalisées par les marchands ou les agents bien que nous ne l'ayons pas vérifié dans ces travaux.

La dernière hypothèse, notée A3, est que l'activité de chacun des acteurs dans le service se restreint à un ou plusieurs ensembles d'acteurs spécifiques avec qui il interagit de manière régulière. Cet ensemble correspond à une communauté d'intérêts, définie par [AKM+05]. Ce concept a également déjà été utilisé pour la détection de fraudes dans le domaine des télécommunications [CPV01].

L'exemple suivant permet d'illustrer cette modélisation des transactions légitimes sous forme d'habitude. La figure 4.3 représente l'espace des transactions possibles considérant l'écart de temps par rapport à la transaction précédente et le montant de la transaction. L'habitude correspond ici à une loi normale à deux dimensions. Le montant suit une loi normale de moyenne 6 et de déviation standard 4 et l'écart de temps entre deux transactions suit une loi normale de moyenne 8 et de déviation standard 2. Les ellipses correspondent à la projection de la fonction de densité de probabilité de l'habitude. Les transactions légitimes générées par le simulateur ont plus tendance à se retrouver au sein de ces ellipses que les transactions frauduleuses.

En 2.1, le champ de l'étude a été restreint aux fraudes comportementales définies par Bhattacharya *et coll.* [BJTW11]. Les attaques modélisées actuellement correspondent à ce type de fraude. La caractéristique principale de ces fraudes est que le comportement du fraudeur et du fraudé se superposent. Le simulateur ajoute donc des comportements frauduleux au comportement générique d'un acteur. Nous adoptons l'hypothèse de Bhattacharya *et coll.* [BJTW11] et supposons que la superposition des comportements normaux et frauduleux implique des ruptures de comportement. Des exemples de ces changements sont illustrés en figure 4.4. La figure 4.4.a) correspond à une variation du montant moyen des transactions et la figure 4.4.b) illustre une modification de la fréquence des transactions. Des parcours fraudeurs particuliers

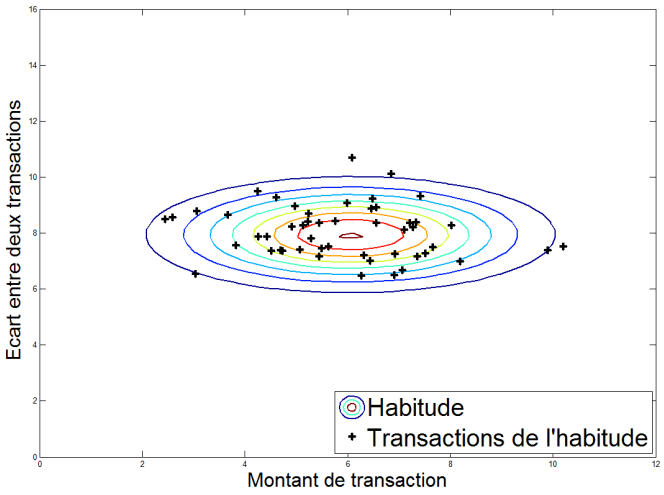
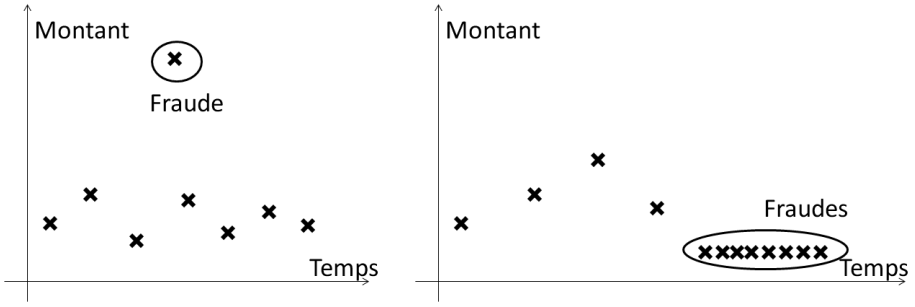


FIGURE 4.3 – Représentation d'une habitude dans l'espace des transactions

sont modélisés et peuvent se produire dans un environnement où les utilisateurs légitimes ont des comportements plus ou moins proches.



(a) Montant moyen (b) Fréquence moyenne

FIGURE 4.4 – Changements des habitudes de paiement

Actuellement, quatre attaques sont modélisées dans le simulateur. Une d'entre elles concerne du blanchiment d'argent, l'autre est un exemple d'attaque lente qui cible un grand nombre d'utilisateurs suite à la compromission de la plateforme de paiement. Ces deux scénarios ont été créés pour le projet MASSIF [Masre]. Ils ne correspondent pas aux fraudes comportementales considérées ici et ne sont donc pas développés dans ce manuscrit.

Les deux dernières attaques sont présentées et étudiées ici. La première est un vol de terminal mobile. L'attaquant peut soit essayer plusieurs fois de deviner le mot de passe de l'utilisateur, avoir observé celui-ci pour voir le code ou tout simplement

en l'obtenant sous la menace. Lorsque le fraudeur est en possession du mot de passe, il réalise plusieurs transactions avec d'autres acteurs du système.

La dernière attaque consiste pour l'attaquant à déployer des programmes malveillants dans les terminaux mobiles afin de créer un réseau de terminaux zombies, en anglais *botnet*. Ces programmes sont ensuite chargés de réaliser des transactions pour le fraudeur à l'insu de l'utilisateur. La modélisation de cette attaque est inspirée de la fraude basée sur le cheval de Troie Zeus, révélé par l'agence fédérale d'investigation américaine *F.B.I* [Fed10], et qui a permis à un réseau international de cybercriminels de dérober 70 millions de dollars en prenant le contrôle de comptes en lignes. Comme dans cette fraude, le programme malveillant modélisé réalise des transactions et envoie la m-monnaie dérobée à des *mules*, également modélisées. Celles-ci sont des porteurs légitimes qui aident sciemment ou non le fraudeur à réaliser une fraude. Comme dans la fraude Zeus, la mule reçoit la transaction frauduleuse et retire l'argent pour l'envoyer en liquide au fraudeur. Contrairement au cas précédent, cette attaque suppose que les cibles disposent d'un terminal sophistiqué permettant l'installation de tels programmes malveillants.

La différence entre la modélisation actuelle et la fraude réelle est que le programme malveillant ne s'adapte pas au comportement de la victime. En effet, d'après le rapport de M86 Security [M8610], dans le cas réel, le programme malveillant intercepte les demandes de transaction de la victime pour modifier le destinataire et le montant. Par contre, dans la modélisation, les transactions réalisées par le programme sont indépendantes de celles réalisées par la victime.

Implémentation

La plateforme de paiement simulée et le modèle comportemental des utilisateurs ont été implémentés sur la plateforme de simulation multi-agent Repast Simphony [NHCV07]. La combinaison des habitudes des différents utilisateurs a été créée grâce à la structure de patron de conception décorateur [GHJV93]. Il s'agit d'une alternative à l'héritage où chaque décoration correspond à une habitude, le tout permettant de construire un comportement complexe.

La figure 4.5 représente deux utilisateurs de profils différents tels qu'ils peuvent être créés avec le patron de conception décorateur [GHJV93]. Le premier acteur réalise habituellement des achats, des transferts et des dépôts. Le deuxième acteur

a l'habitude de réaliser des dépôts, des retraits et des transferts. Les différentes habitudes peuvent être paramétrées. Il est donc possible de modéliser un premier acteur qui effectue des transferts mensuels et un second qui effectue des transferts hebdomadaires.

(a) Acteur 1

(b) Acteur 2

FIGURE 4.5 – Résultat du patron de conception décorateur

4.1.3 Validation préliminaire

L'évaluation du prototype de générateur de données passe par celle de ses différents composants : la plateforme simulée et le modèle comportemental des habitudes. La modélisation de la plateforme et son implémentation ont été validés au sein d'Orange Labs. La validation du modèle comportemental des utilisateurs, est basée sur l'évaluation de l'hypothèse A2. Pour cela, des données collectées sur neuf mois de fonctionnement d'un système de paiement opérationnel où plusieurs centaines de milliers d'utilisateurs ont effectué plusieurs millions de transactions. Le jeu de données a été nettoyé afin d'en faciliter l'analyse. Ainsi, les transactions enregistrées en double ont été supprimées. Toutes les transactions ayant échoué ont également été écartées. La notion d'habitude est étudiée pour les porteurs uniquement, seules les transactions impliquant un porteur sont considérées.

L'hypothèse A2 implique que les porteurs ont des habitudes de paiement qui suivent une loi normale à deux dimensions, les montants et les écarts de temps entre deux transactions. Afin d'évaluer la validité de cette hypothèse, nous avons séparé l'ensemble des transactions réalisées par chacun des utilisateurs et les avons

regroupées par type de transaction réalisée. Nous obtenons ainsi 638 306 listes de transactions. Chacune est liée à un couple (utilisateur, type de transaction) unique que nous désignons par le terme activité. Pour récapituler, une activité correspond à l'ensemble des transactions d'un certain type réalisées par un certain utilisateur.

Parmi les 638 306 activités recensées, 22 362, soit 3,5%, correspondent à plus de 30 transactions. Celles-ci appartiennent à 4 939 porteurs parmi 374 130, ce qui représente environ 1,32% des porteurs. Ces activités ont été conservées pour réaliser un test du χ^2 [Mat] sur les montants et les écarts entre deux transactions. Il s'agit d'un test statistique permettant d'évaluer l'adéquation entre une loi de probabilité et une série de données.

Nous avons choisi de ne pas considérer l'ensemble des activités pour plusieurs raisons. Tout d'abord, celles composées de 1 à 5 transactions représentent près de 50% des comportements. Celles-ci peuvent être facilement modélisées par le simulateur mais ne nous semblent pas suffisemment volumineuses pour qu'il s'agisse d'une habitude d'un utilisateur.

Ce grand nombre d'activités de petite taille peut s'expliquer par le fait que les enregistrements correspondent à une phase de développement du service. Il est donc possible que certaines de ces activités correspondent à des débuts d'habitude qui n'ont pas encore atteint un rythme de croisière ou à des utilisateurs qui ont utilisé le service et l'ont abandonné, par exemple. La deuxième raison est que la probabilité qu'un test du χ^2 soit un faux positif est plus élevée si la série de données est de faible taille. Afin d'éviter ces biais, nous n'avons considéré que les activités constituées d'un nombre suffisant de transactions. Nous avons fixé cette limite à 30 mais l'étude pourrait être réalisée avec des activités de 10 ou 20 transactions.

Comme le récapitule la table 4.1, environ 23,61% des activités considérées respectent la représentation statistique que nous avons définie pour une habitude. Il serait cependant faux de conclure que les autres activités ne sont pas des habitudes. En effet, l'hypothèse testée ici est qu'une activité correspond à une habitude particulière. Pourtant, il est possible qu'une activité soit composée de deux habitudes. Par exemple, un porteur peut avoir l'habitude de réaliser des achats alimentaires, un paiement marchand d'un montant proche de 50 m-monnaie toutes les semaines et d'aller au cinéma, autre paiement marchand pour 10 m-monnaie tous les mois.

Ces résultats préliminaires permettent de valider en partie les hypothèses qui

TABLE 4.1: Résultats du test pour détecter des habitudes parmi les activités

	Nombre d'activités	Pourcentage d'activités
Montant et période suivent une distribution normale	5 280	23,61%
Montant ou période suivent une distribution normale	9 876	44,16%
Ni montant ni période ne suivent une distribution normale	7 206	32,22%

sont à l'origine du simulateur décrit ci-dessus. La configuration utilisée dans le cadre de cette thèse est décrite dans la partie suivante. Bien que les hypothèses A1 et A3 nous paraissent bonnes, celles-ci restent à valider.

4.1.4 Configuration et génération de jeux de données synthétiques

Les données produites par le simulateur dépendent des paramètres des différents acteurs modélisés. Nous décrivons ici comment chaque type d'acteur a été paramétré à partir de l'étude des données réelles. L'objectif est de comprendre les données manipulées dans la section 4.2 concernant l'adaptation de méthodes de classification à la détection de la fraude.

Porteurs légitimes

Pour que les comportements des porteurs modélisés soient aussi complexes que ceux des utilisateurs réels, différentes habitudes leur sont attribuées. Les paramètres de celles-ci sont basés sur plusieurs statistiques exposées ici. Celles-ci sont calculées à partir des activités qui vérifient la notion d'habitude.

Tout d'abord, les proportions d'habitudes présentes parmi les populations sont étudiées. Ces mesures permettent de mieux connaître les différents types de porteurs à modéliser et d'en dresser un profil. Le profil des utilisateurs est ensuite précisé par le calcul des paramètres des lois statistiques représentant les montants et les écarts de temps entre deux transactions.

Les porteurs arborent entre 1 et 4 habitudes différentes. La table 4.2 regroupe les pourcentages de porteurs correspondant à chaque nombre d'habitudes. Une grande majorité des porteurs de la population sélectionnée n'a qu'une seule habitude tandis qu'un quart d'entre eux en a deux. Enfin, environ 9% d'entre eux ont trois habitudes de types différents. Finalement, seuls 1,86% d'entre eux ont quatre habitudes.

TABLE 4.2: Répartition du nombre d'habitudes

Nombre d'habitudes	Proportion de porteurs
1	63,17%
2	26,30%
3	8,67%
4	1,86%

La table 4.3 représente la répartition des catégories de transactions par rapport au nombre d'habitudes du porteur. Pour ceux qui ont une seule habitude, l'achat de temps de communication pour leur terminal mobile est le plus représenté avec 82,69% et le moins représenté est l'achat marchand avec 0,22% des comportements. Pour les porteurs ayant deux habitudes, les plus rencontrées sont le dépôt et l'achat de temps de communication puisqu'ils correspondent à respectivement 82,37% et 60,35% des porteurs. L'habitude de paiement marchand est présent chez 2,46% des porteurs. De même, pour les porteurs arborant 3 habitudes, les plus représentées sont le dépôt et l'achat de temps de communication. Par contre, l'achat auprès de marchand est l'habitude la moins représentée. Finalement, le paiement marchand est l'habitude la moins rencontrée auprès des porteurs qui ont 4 habitudes. Tous les porteurs de cette catégorie réalisent des transferts.

TABLE 4.3: Répartition du nombre d'habitudes

	1 habitude	2 habitudes	3 habitudes	4 habitudes
Dépôt	11,54%	82,37%	97,66%	98,91%
Retrait	2,76%	18,86%	46,73%	97,83%
Paiement marchand	0,22%	2,46%	3,50%	6,52%
Transfert entre particuliers	2,79%	35,95%	63,55%	100%
Achat temps de communication	82,69%	60,35%	88,55%	96,74%

Les différentes caractéristiques listées ci-dessus permettent de créer les porteurs et de respecter la proportion des habitudes qu'ils possèdent. Pour modéliser un porteur

dans le simulateur, un comportement comprenant un certain nombre d'habitudes, lui est attribué. Afin de paramétrer ces différentes habitudes, un montant moyen, un écart-type de montant, une fréquence moyenne et un écart-type de fréquence sont attribués à chacune d'elles. Chaque paramètre est défini en suivant une loi normale dont les caractéristiques sont présentées dans les tableaux 4.4 et 4.5.

En observant ces deux tableaux, on constate que l'écart-type peut être supérieur à la moyenne. Il s'agit d'un signe de surdispersion et donc d'une hétéréogénéité entre les caractéristiques des individus.

TABLE 4.4: Statistiques concernant les montants en m-monnaie

	Montant moyen		Écart-type de montant	
	Moyenne	Ecart-type	Moyenne	Écart-type
Dépôt	46 013,46	97 766,61	62 784,05	117 976,93
Retrait	86 127,02	151 837,87	79 133,20	94 122,82
Paiement marchand	47 634,94	51 969,88	48 897,34	66 762,42
Transfert entre particuliers	32 792,20	52 115,02	41 306,16	56 935,23
Achat temps de communication	1 420,09	2 454,68	1 100,26	1 870,96

TABLE 4.5: Statistiques concernant les périodes ou écarts de temps entre deux transactions en jours

	Période moyenne		Ecart-type de période	
	Moyenne	Ecart-type	Moyenne	Ecart-type
Dépôt	5,27	2,21	6,98	3,57
Retrait	5,46	2,05	6,99	3,82
Paiement marchand	5,29	2,30	6,67	3,86
Transfert entre particuliers	4,07	2,12	6,58	4,45
Achat temps de communication	3,82	2,13	8,45	5,92

Ces différentes mesures ont été utilisées pour générer les profils de 2 000 porteurs dans la simulation. D'autres paramètres pourraient être pris en compte, comme la date de début ou de fin d'un comportement ou la taille des communautés d'intérêt d'un porteur. Ce dernier paramètre correspond au nombre de marchands, agents, particuliers avec lesquels un porteur réalise régulièrement des transactions. A ce stade, ces paramètres sont pris en compte dans la simulation mais ne sont pas dérivés des données réelles. Ainsi, la date de début des comportements est choisie aléatoirement

au sein du premier mois de simulation et les comportements se déroulent tout au long de la simulation. Finalement, la taille des différentes communautés d'intérêt est choisie aléatoirement entre 1 et 10.

Acteurs frauduleux

Pour l'étude réalisée dans cette thèse, trois sortes d'acteurs frauduleux ont été mis en place, des voleurs, des agents, *bots* qui infectent le terminal mobile et réalisent des transactions à l'insu de leur hôte et des mules qui permettent aux maîtres de bots de récupérer leur butin.

Les mules se contentent de retirer le montant correspondant à une transaction frauduleuse. Ils sont donc associés à peu de paramètres dans la simulateur. Les seuls sont le délai séparant la réception de la fraude et la date de retrait. Ce délai est choisi aléatoirement et est d'au plus 2 jours. Le nombre de ces fraudes devrait donc être équivalent à celui des transactions réalisées par les fraudeurs. Nous supposons que la proportion de porteurs qui sont des mules est très faible. Nous avons choisi une proportion de 0,2% dans la simulation.

La fraude liée au botnet Zeus s'adaptait au comportement de l'utilisateur pour éviter d'alerter les organismes qui géraient les comptes impactés [M8610]. Cette attaque est plutôt lente et son montant est plutôt inférieur à celui des transactions réalisées par la victime. Pour représenter cela, dans la simulation, la fréquence de ce comportement correspond au double de la fréquence moyenne observée pour les transferts entre particuliers. Le montant de la fraude suit une loi normale dont la moyenne μ et l'écart-type σ suivent les formules 4.1 et 4.2 :

$$\mu = \frac{\mu\_M^T - \sigma\_M^T}{2} \tag{4.1}$$

$$\sigma = \frac{\mu\_S^T - \sigma\_S^T}{2} \tag{4.2}$$

où μ_M^T et σ_M^T correspondent respectivement à la moyenne et l'écart-type des montants moyens des transferts entre particuliers et où μ_S^T et σ_S^T correspondent respectivement à la moyenne et l'écart-type de l'écart-type moyen des transferts.

Les voleurs constituent 0,15% de la population qui effectuent un vol tous les 1 à 2 jours. Chacun d'entre eux cible une vingtaine de porteurs. Nous considérons que l'intérêt de ces fraudeurs est de retirer une certaine quantité d'argent le plus rapidement possible sans que les retraits soient bloqués. Le montant de la fraude est

choisi de la même manière que décrit ci-dessus. De plus, la fréquence des transactions est élevée et concentrée dans une courte période de temps qui suit le vol.

Marchands et agents de distribution de m-monnaie

Actuellement, le comportement actif des marchands et des agents de distribution n'a pas été modélisé. Ces derniers sont capables de réaliser des opérations demandées par un client mais ne déploient pas de stratégie plus avancée comme la gestion de leur stock de monnaie électronique ou de monnaie fiduciaire. Leur nombre est la seule caractéristique à paramétrer actuellement. La proportion des marchands et des agents est respectivement de 0,08% et de 5,80%.

4.1.5 Discussion concernant la génération de données artificielles

Dans cette section, la modélisation du système de transactions sur mobile et de ses utilisateurs ainsi que le simulateur développé pour générer des données synthétiques ont été décrits. Le comportement des porteurs est celui qui a été modélisé le plus finement dans la version actuelle. Les marchands et les agents qui distribuent la m-monnaie réalisent les transactions mais ne réalisent pas d'action active et ne mettent pas en oeuvre de stratégies pour gérer leur commerce. Dans le monde réel, comme l'indiquent Jack *et coll.* [JTT10], ils doivent gérer leur stock de m-monnaie et de monnaie fiduciaire.

Une validation préliminaire de cette modélisation a également été réalisée. Celle-ci a démontré que le concept d'habitude est valide en soumettant plusieurs activités à un test du χ^2. Ce test a permis de montrer qu'au moins 23,61% des activités correspondent à une habitude. Le concept d'habitude peut être étendu pour prendre en compte le fait qu'un porteur peut avoir plusieurs habitudes pour un type de transactions donné. En effet, nous avons exposé un cas où l'activité d'un porteur pourrait être subdivisée en plusieurs activités qui suivent la distribution statistique d'une habitude.

Finalement, la configuration utilisée pour générer les données utilisées dans notre étude de l'adaptation des algorithmes de classification a été présentée. L'analyse de données réelles a permis de calculer les paramètres des porteurs légitimes. Pour les fraudeurs, nous avons décidé un parcours et une stratégie particulière. La configuration présentée ici est une des configurations possibles et est le résultat de choix que

nous avons réalisés. D'autres parcours fraudeurs sont possibles.

La représentativité des données sélectionnées pour la validation du modèle et la configuration, que nous voulions statistiquement réaliste, peuvent être discutées. En effet, comme le montre la table 4.6, la répartition des types de comportement dans la population sélectionnée est plutôt bien respectée pour chaque type de comportement mis à part le dépôt et l'achat de temps de communication. Ceci est dû au fait que seuls les comportements de plus de 30 transactions ont été retenus pour l'étude. En effet, on peut supposer, intuitivement, que les utilisateurs réalisent un dépôt pour plusieurs achats. Ce type de comportement est alors défavorisé dans la sélection. Cette hypothèse est confirmée par l'observation des données puisque 97,40% des comportements de type dépôt sont composés de moins de 30 transactions alors que 84,04% des comportements de type achat de temps de communication comprennent moins de 30 transactions.

TABLE 4.6: Proportions de porteurs qui arborent un certain type de comportement

Décoration	Proportion de la population totale	Proportion de la population sélectionnée
Dépôt	98,20%	39,26%
Retrait	23,65%	12,57%
Paiement marchand	1,40%	1,21%
Transfert entre particuliers	19,10%	18,59%
Achat temps de communication	27,05%	77,59%

Après la description du simulateur de données et de la configuration utilisée dans le cadre de la thèse, la seconde contribution de la thèse est maintenant détaillée. Celle-ci a pour but d'évaluer plusieurs algorithmes de classification et de les adapter pour la détection de la fraude dans notre contexte.

4.2 Adaptation d'algorithmes de classification

En section 2.3, nous avons montré que différentes méthodes de classification peuvent être appliquées pour la détection de fraude dans les systèmes de transactions sur terminaux mobiles et qu'une phase d'adaptation est nécessaire. L'objectif ici est de présenter une telle approche ainsi qu'une évaluation préliminaire des algorithmes de classification pour la détection de fraude dans les systèmes de transactions sur terminaux mobiles. Cette étude se limite aux algorithmes de classification avec un

modèle construit par un apprentissage automatique supervisé ou par l'étude d'autres instances.

Comme indiqué en section 2.3, il n'existe pas à notre connaissance d'étude publique concernant l'adaptation des méthodes de classification pour la détection de la fraude dans les systèmes de transactions sur terminaux mobiles. Les études sur les algorithmes de classification pour la détection de fraude qui existent concernent les domaines bancaires [PWKS11] et des télécommunications [BH01, FP97, HM08]. Parmi les sept bases de données mentionnées par Peng *et coll.* [PWKS11], seule la base *UCI MLR* qui concerne les demandes de crédit pour un paiement par carte est accessible aujourd'hui. De plus, certains champs qui la composent comme le mode de financement du lieu de résidence, location, possession ou en cours de remboursement ou la durée d'emploi n'évoluent pas forcément à chaque transaction et semblent donc avoir peu d'impact sur la recherche de modifications de comportement. Ceci met en évidence les différences qui peuvent exister entre différentes formes de paiement et donc le besoin de réaliser une étude spécifique à notre domaine.

L'objectif de cette section est d'étudier l'adaptation des méthodes de classification. La méthodologie est d'abord détaillée et les résultats des différentes études sont présentés. Finalement, une discussion conclut cette section.

4.2.1 Méthodologie

L'étude présentée ici se déroule en quatre étapes. Tout d'abord, différentes représentations des données à classer sont comparées. Celles qui donnent les meilleurs résultats seront ensuite sélectionnées pour la deuxième étape de cette étude, à savoir la comparaison des algorithmes de classification. Les algorithmes présentant les meilleurs résultats sont ensuite sélectionnés. Les différents paramètres de ces derniers algorithmes sont ensuite optimisés. L'étude est poursuivie par l'utilisation de l'algorithme et des paramètres optimum choisis pour détecter les fraudes dans une nouvelle base de données.

Dans la prochaine partie, la méthodologie utilisée pour réaliser ces trois phases d'évaluation est détaillée. Tout d'abord, les différents critères d'évaluation utilisés sont décrits ainsi que les différents jeux de données considérés. Finalement, la procédure de chacune des trois expériences est détaillée.

Critères d'évaluation

La sous-représentativité de certaines classes dans les données à étudier est à prendre en compte pour le choix des critères d'évaluation des algorithmes de classification. En effet, dans le cas où une ou plusieurs classes sont rares, certains critères peuvent révéler des résultats inexacts [Cha05]. Considérons, par exemple, un algorithme qui classe en tant que transactions légitimes toutes les entrées d'une base de données qui contient 99% de transactions légitimes et 1% de transactions frauduleuses. Dans ce cas, si le critère d'évaluation retenu est le taux de classification correcte, alors le résultat sera 99% de réussite. Pourtant, aucune transaction frauduleuse n'a été correctement classifiée.

Pour éviter ce biais, en plus du taux de classifications correctes, nous avons choisi comme critères d'évaluation le taux de classification correcte ainsi que le coefficient Kappa [Coh60] et le coefficient de corrélation Matthews [Mat75]. Ces indicateurs permettent de mesurer la pertinence des algorithmes de classification lorsque la répartition des classes est mal équilibrée. Le temps d'exécution est aussi pris en compte pour évaluer les algorithmes. Pour cela, la durée de la phase de test et la durée de la phase d'apprentissage sont considérées. Ces différents critères ont été utilisés dans les quatre étapes de l'étude.

Le taux de classification correcte mesure le pourcentage des transactions qui ont été correctement labellisées. Plus cette valeur est proche de 100%, meilleure est la performance de l'algorithme. Ce taux est conservé à titre indicatif.

Le coefficient de corrélation Matthews [Mat75] est une manière de synthétiser la matrice de confusion. Il s'agit d'une matrice carrée où les lignes et les colonnes représentent les différentes classes considérées. Chaque case de la matrice correspond au nombre d'instances de la classe réelle (représentée sur la ligne) qui ont été classifiées en tant que la classe représentée en colonne. La diagonale de cette matrice correspond donc au nombre d'instances classifiées correctement. La matrice de confusion d'un problème de classification à 2 classes où les transactions frauduleuses représentent la classe positive et les transactions légitimes représentent la classe négative est représentée par la table 4.2.1.

Le coefficient de Matthews de cette matrice correspond à :

$$\frac{VP.VN - FP.FN}{\sqrt{(VP+FP)(VP+FN)(VN+FP)(VN+FN)}} \tag{4.3}$$

Les valeurs obtenues sont comprises entre -1 et 1. La valeur -1 correspond à un

TABLE 4.7: Matrice de confusion d'un problème de classification à 2 classes

		Prédit	
		Légitime	Frauduleux
Réel	Légitime	Nombre de vrais négatifs (VN)	Nombre de faux positifs (FP)
	Frauduleux	Nombre de faux négatifs (FN)	Nombre de vrais positifs (VP)

désaccord total entre les variables prédites et la vérité terrain. La valeur 0 indique que l'algorithme de classification est équivalent à une fonction aléatoire et la valeur 1 correspond à une prédiction parfaite.

Le coefficient Kappa [Coh60], permet de mesurer à quel point un algorithme de classification est différent d'une décision prise au hasard. La formule permettant de calculer le coefficient Kappa est :

$$\frac{Pr(a) - Pr(e)}{1 - Pr(e)} \tag{4.4}$$

où $Pr(a)$ représente la probabilité que l'algorithme de classification étudié soit en accord avec la vérité terrain et $Pr(e)$ représente la probabilité qu'une fonction aléatoire soit en accord avec la vérité terrain. Ces deux proportions sont également calculées à partir de la matrice de confusion. La proportion $Pr(a)$ est calculée à partir de la formule :

$$Pr(a) = \frac{VP + VN}{VP + VN + FP + FN} \tag{4.5}$$

La proportion $Pr(e)$ est calculée à partir de la formule :

$$Pr(e) = \frac{(VN + FP)(VN + FN) + (VP + FN)(VP + FP)}{(VP + VN + FP + FN)^2} \tag{4.6}$$

Plus Kappa est proche de 1 et moins l'algorithme de classification ressemble à une fonction de décision aléatoire. De plus, si l'algorithme de classification classifie très mal les données, $Pr(a)$ est proche de 0 et Kappa est donc très proche de 0 ou même négatif. Landis et Koch [LK77] ont proposé la table 4.8 pour interpréter les différentes valeurs de Kappa.

La durée des phases d'apprentissage et de tests permettent de connaître les performances en terme de temps de calcul des différents algorithmes. Les temps de calcul les plus faibles sont préférés ici.

TABLE 4.8: Interprétation du coefficient Kappa selon Landis et Koch, source : [LK77]

Valeurs de Kappa	Interprétation
< 0	Désaccord
$0 - 0,20$	Accord très faible
$0,21 - 0,40$	Accord faible
$0,41 - 0,60$	Accord modéré
$0,61 - 0,80$	Accord fort
$0,81 - 1$	Accord presque parfait

Jeux de données utilisés

À partir de la configuration détaillée en section 4.1.4, deux jeux de données, A et B, ont été créés pour une durée de 4 mois chacun. Ils mettent en œuvre 2000 porteurs, 2 marchands, 6 agents, 40 terminaux infectés et 3 voleurs. La base A contient 54 848 transactions. Parmi elles, 53 170 sont légitimes, 240 transactions suivent un vol, 721 sont envoyées par un logiciel zombie vers une mule et 717 correspondent aux retraits effectués par les mules. Le montant moyen des transactions légitimes est 56 494 m-monnaie. Le montant moyen des transactions suivant les vols est 13 001 m-monnaie et le montant moyen des transactions liées au réseau de logiciel zombie est de 10 690 m-monnaie. Quant à la base B, elle contient 54 224 transactions. Parmi elles, 52 532 sont légitimes, 232 transactions suivent un vol, 731 sont envoyées par un logiciel zombie vers une mule et 729 correspondent aux retraits effectués par les mules. Le montant moyen des transactions légitimes est 54 311 m-monnaie. Le montant moyen des transactions suivant les vols est 13 421 m-monnaie et le montant moyen des transactions liées au réseau de logiciel zombie est de 10 816 m-monnaie.

Plusieurs pré-traitements ont été réalisés à ces deux bases de données afin de créer des représentations différentes. Quinze nouvelles bases de données ont été créées à partir de chacun des jeux de données de départ. Pour plus de facilité, seules les pré-traitements affectant la base A sont détaillées ici. Les mêmes traitements ont été appliqués à la base B.

Tout d'abord, le format des données représenté en 4.9 est conservé et seule la représentation de la vérité terrain est altérée. Au départ, la vérité terrain est représentée par huit classes. Cette base de données initiale est notée A8. Les classes légitimes sont : N-RegDep les habitudes de dépôt, N-RegRC les habitudes d'achat de temps de communication, N-RegC2C les transferts entre particuliers légitimes, N-RegWith les retraits légitimes sous forme d'habitude et N-ServMerch les habitudes

qui correspondent à un paiement marchand. Les classes qui représentent les transactions frauduleuses sont : F-SevWith les transactions réalisées suite au vol d'un terminal, F-bot les transferts effectués d'un téléphone infecté vers une mule et F-MuleWith un retrait effectué par une mule. Pour le premier traitement réalisé, toutes les classes correspondant à des événements légitimes sont regroupées en une seule classe pour obtenir le jeu de données A4 qui contient 4 classes différentes. Ensuite, deux classes qui correspondent à deux phases d'une même fraude sont regroupées. Le jeu de données A3 est ainsi créé. Finalement, dans le jeu de données A2, les transactions sont sont qualifiées de normales ou frauduleuses sans aucune autre considération.

Table 4.9: Format brut

Champ	Signification
S_t	Type de transaction réalisée
T_a	Montant de la transaction
S_PRE_BAL	Solde de l'expéditeur avant la transaction
S_POST_BAL	Solde de l'expéditeur après la transaction
R_PRE_BAL	Solde du destinataire avant la transaction
R_POST_BAL	Solde du destinataire après la transaction
S_S	État du compte de l'expéditeur
R_S	État du compte du destinataire
S_cc	Catégorie de l'expéditeur
R_cc	Catégorie du destinataire
date	Date
mois	Mois
annee	Année
heure	Heure
minute	Minute
seconde	Seconde
ver_ter	Vérité terrain

À partir de ces quatre bases dont la représentation de la vérité terrain est modifiée, le format des données est modifié et certains indicateurs correspondant à différentes agrégations de données sur une certaine période sont ajoutés. Les différents champs du format de départ et du format modifié sont présentés respectivement dans les tables 4.9 et 4.10. L'ordre d'apparition des champs dans les tables correspond à leur ordre dans les enregistrements. Les données agrégées ne prennent en compte que les événements qui précèdent la transaction en cours. Quatre nouvelles bases, A8_modif,

A4_modif, A3_modif, A2_modif sont ainsi créées.

Le format brut décrit la transaction uniquement. Il contient différents champs qui indiquent le montant, la date, l'expéditeur et le destinataire de la transaction ainsi que leurs soldes respectifs avant et après la transaction.

Whitrow *et coll.* [WHJ+09] suggèrent que les performances des algorithmes de classification pour la détection de la fraude sont meilleures lorsque les données en entrée comprennent des données agrégées. Cette préconisation a été prise en compte pour le format modifié. Celui-ci comprend plusieurs champs où des informations sont agrégées sur une certaine durée. Ainsi, les champs hour_nb_transactions, day_nb_transactions et week_nb_transactions correspondent au nombre de transactions réalisées par le porteur à l'origine de la transaction respectivement pendant l'heure, le jour et la semaine qui précède la transaction. De cette manière, le format modifié contient des informations concernant la transaction mais il la replace également dans l'historique du porteur à l'origine de la transaction.

À ce stade, il existe en tout huit jeux de données. À chacun d'entre eux est appliquée une analyse en composantes principales [AW10]. Celle-ci a pour objectif de décorréler les différentes dimensions, c'est-à-dire les champs des formats détaillés ci-dessus, qui caractérisent un événement. Les informations importantes sont extraites des données de départ et représentées dans un espace dont les vecteurs directeurs sont orthogonaux. Cela permet de réduire la dimension des événements considérés tout en conservant les informations les plus pertinentes. Il en résulte 8 bases dont le nom suit la forme AX_PCA ou AX_modif_PCA selon que les données traitées sont les données brutes ou modifiées. Dans cette forme, X correspond aux nombres de classes représentées dans la vérité terrain.

Expérimentations

Les différentes expérimentations ont été menées à l'aide du logiciel WEKA, *Waikato Environment for Knowledge Analysis* [HFH+09]. Celui-ci regroupe plusieurs implémentations d'algorithmes de fouille de données et d'apprentissage automatique. Une bibliothèque d'algorithmes est disponible pour le développement ainsi qu'une interface graphique qui permet de les manipuler.

La première expérimentation menée consiste à sélectionner les meilleures représentations des données parmi celles détaillées en 4.2.1. Pour cela, trois algorithmes :

TABLE 4.10: Format modifié

Champ	Signification
S_t	Type de transaction réalisée
T_a	Montant de la transaction
S_cc	catégorie de l'expéditeur
R_cc	catégorie du destinataire
week_nb_transactions	Nombre de transactions de la semaine
week_min_amount	Plus petit montant de la semaine
week_max_amount	Plus grand montant de la semaine
week_mean_amount	Montant moyen de la semaine
week_total_amount	Montant total de la semaine
week_nb_transactions_with_contact	Nombre de transactions de la semaine avec ce partenaire de transaction
day_nb_transactions	Nombre de transactions de la journée
day_min_amount	Plus petit montant de la journée
day_max_amount	Plus grand montant de la journée
day_mean_amount	Montant moyen de la journée
day_total_amount	Montant total dépensé de la journée
day_nb_transactions_with_contact	Nombres de transactions de la journée avec ce partenaire
hour_nb_transactions	Nombre de transactions dans l'heure
hour_min_amount	Montant minimal des transactions réalisées dans l'heure
hour_max_amount	Montant maximal des transactions réalisées dans l'heure
hour_mean_amount	Montant moyen des transactions réalisées dans l'heure
hour_total_amount	Montant total des transactions réalisées dans l'heure
hour_nb_transactions_with_contact	Nombre de transactions réalisées dans l'heure avec ce partenaire
delay_with_previous	Ecart de temps avec la transaction précédente
day_of_week	jour de la semaine
day_of_month	jour du mois
datetime_hour	heure
datetime_minute	minute
datetime_totalseconds	seconde
ver_ter	Vérité terrain

une forêt aléatoire [Bre01], un réseau de neurones de type perceptron [WL90] et une régression logistique linéaire [LHF05], sont appliqués aux différents jeux de données dérivés de la base de données A. L'utilisation de trois algorithmes permet à la fois d'obtenir un consensus sur plusieurs méthodes et de restreindre les temps de calculs

et de performances de l'expérimentation.

Les paramètres des algorithmes sont ceux proposés par défaut par le logiciel. La méthode de la validation croisée est utilisée. Celle-ci consiste à échantillonner plusieurs fois le jeu de données considéré, ici 10 fois. Chaque échantillon est ensuite utilisé en tant que base de test face aux autres échantillons qui servent à l'apprentissage. L'objectif est de fiabiliser l'apprentissage d'un modèle sur un ensemble de données. Les données qui représentent les meilleurs résultats au regard des critères définis en 4.2.1 sont sélectionnées pour les expérimentations suivantes.

La seconde expérimentation a pour but de sélectionner les meilleurs algorithmes de classification à base d'un modèle déterminé par un apprentissage automatique supervisé comme défini en section 2.3. Des algorithmes à base d'un modèle défini par l'étude d'autres instances présentés en section 2.3 sont également pris en compte.

Cette étude est basée sur douze algorithmes qui représentent différentes catégories de méthode de classification définies en section 2.3 et implémentés dans WEKA. Ces algorithmes sont : un réseau bayésien [Bou04] ; une classification bayésienne naïve [JL95] ; un séparateur à vaste marge [HDO+98] ; une régression logistique multinomiale avec une estimation de type ridge [LCVH92] ; un arbre de régression logistique [LHF05] ; un réseau de neurones de type perceptron [WL90] ; la méthode des k plus proches voisins [AKA91] ; l'algorithme K* [CT95] ; une table de décision majoritaire [Koh95] ; une table de décision de type PART qui est associée à un classifieur C4.5 [FW98] ; un arbre de décision C4.5 [Qui93] et une forêt aléatoire [Bre01]. Les paramètres par défaut sont utilisés. Les critères définis en 4.2.1 permettent d'évaluer ces algorithmes et de choisir ceux qui présentent les meilleures performances. Seules les bases sélectionnées dans l'étape précédente sont utilisées ainsi que la méthode de validation croisée.

Les deux premières étapes de l'expérimentation sont basées sur l'utilisation des paramètres par défaut proposés par le logiciel. Elles ont permis de sélectionner les jeux de données et les algorithmes les plus performants pour le problème considéré ici. Ces méthodes sont optimisées dans une troisième éxpérimentation. Pour cela, chaque algorithme sélectionné précédemment en faisant varier les paramètres pour classifier les données. Cette étude est encore réalisée avec les variations des jeux de données A sélectionnées à la première étape. Les résultats et mesures liées à l'application des différents paramètres sont évalués grâce aux critères définis ci-dessus.

Les différents choix réalisés ci-dessus sont ensuite validés au cours d'une quatrième expérimentation. Les algorithmes adoptés et leurs paramètres optimum réalisent leur apprentissage sur les variations de la base A sélectionnées et réalisent les tests sur les mêmes variations de la base B. Si les résultats d'évaluation ainsi obtenus sont satisfaisants, cela signifie que les algorithmes optimisés sélectionnés répondent bien au problème et ne sont pas sujets à du sur-apprentissage, pour lequel une règle est créée pour chaque cas particulier, ni au sous-apprentissage, pour lequel les fraudes qui sont la classe minoritaire seraient beaucoup plus sujets aux erreurs de classification. De plus, ils sont applicables à des bases de données ayant des caractéristiques similaires à celles des ensembles de transactions pris en compte ici.

4.2.2 Résultats

Les résultats des différentes expérimentations sont exposés ici.

Sélection des jeux de données

Le tableau 4.11 regroupe les résultats des tests sur les différents jeux de données. Pour chaque algorithme et chaque base de données, le pourcentage de classification correcte, le coefficient Kappa, le coefficient de corrélation Matthews, le temps d'apprentissage et le temps de tests sont mesurés. La moyenne de chacun de ces indicateurs sur les trois algorithmes de classification est calculée et représentée dans le tableau 4.11.

Les jeux de données dont huit classes sont distinguées dans la vérité terrain, de type A8 ou B8, sont ceux qui affichent les meilleures performances de classification. Cependant, leur durée d'apprentissage est plus élevée que les autres algorithmes. Les données qui ont subi l'analyse en composantes principales donnent de moins bons résultats que les autres représentations de données. Les données brutes ayant subi l'analyse en composantes principales sont celles qui présentent les résultats les moins bons. En particulier, la base A2_PCA affiche un coefficient Kappa de 0,39 et un coefficient de corrélation Matthews de 0,39. Les données ne peuvent donc pas être facilement décorrélées. Les bases AX_modif, où X correspond au nombre de classes représentées dans la vérité terrain, affichent de meilleures performances que les autres représentations. Cela implique que l'agrégation du montant et du nombre de transactions sur différentes échelles de temps permet de mieux discriminer les

TABLE 4.11: Comparaison des représentations de données, moyenne des indicateurs. Les valeurs de Kappa et Matthews supérieures à 0,7 sont signalées par l'astérisque *.

	Valeurs moyennes				
	Taux de réussite	Kappa	Matthews	Apprentissage (sec)	Test (sec)
A2	98.16	0,71*	0,71*	55,63	0,01
A3	98.05	0,63	0,65	140,95	0,03
A4	97.27	0,63	0,65	171,75	0,03
A8	98.18	0,97*	0,98*	265,58	0,05
A2_modif	98.51	0,72*	0,72*	157,46	0,03
A3_modif	98.44	0,71*	0,72*	187,09	0,03
A4_modif	98.46	0,72*	0,72*	309,08	0,05
A8_modif	98.59	0,98*	0,98*	310,19	0,04
A2_PCA	97.47	0,39	0,39	35,50	0,02
A3_PCA	97.47	0,44	0,49	89,16	0,03
A4_PCA	97.67	0,55	0,58	66,89	0,02
A8_PCA	97.53	0,96*	0,97*	129,16	0,04
A2_modif_PCA	98.36	0,68	0,69	40,06	0,02
A3_modif_PCA	98.27	0,67	0,69	48,01	0,02
A4_modif_PCA	98.35	0,68	0,70*	61,26	0,02
A8_modif_PCA	98.26	0,97*	0,98*	216,92	0,07

transactions frauduleuses des transactions légitimes.

Notons que le tableau met en évidence le fait que le taux de classifications réussies n'est pas un critère pertinent pour les problèmes où une classe est sous-représentée puisque la base A2_PCA affiche un taux de réussite de 97,47% alors que les faibles valeurs des indicateurs Kappa et Matthews indiquent que les performances de classification sur cette base sont médiocres.

Pour les expériences suivantes, la base A8_modif est sélectionnée puisqu'elle présente les meilleures performances de classification. Cependant, il ne semble pas évident que dans la réalité, les comportements légitimes soient décomposés pour mettre en évidence les habitudes des utilisateurs. La base A2_modif, qui est la meilleure base parmi celles qui ne distinguent pas les habitudes, est également sélectionnée pour la suite des expérimentations.

Sélection des algorithmes

Suite à la première expérience, les bases A2_modif et A8_modif ont été sélectionnées. Douze algorithmes de classification leur ont été appliqués. Les différents indicateurs mesurés sont regroupés dans le tableau 4.12 pour la base A2_modif et dans le tableau 4.13 pour la base A8_modif.

Pour la base A2_modif, le séparateur à vaste marge, *SVM*, est l'algorithme le moins performant. En effet, sa phase d'apprentissage est très longue, 2 773,42 secondes et ses coefficients de Kappa et Matthews sont nuls. La méthode bayésienne naïve est également très peu performante puisque même son taux de classification correcte, 47,72%, est très faible comparé aux autres algorithmes. Le réseau bayésien, la méthode des k-plus proches voisins, K* affichent des performances similaires avec des coefficients de Kappa et Matthews qui ont une valeur comprise entre 0,5 et 0,6. Pour les algorithmes régression logistique munltinomiale, régression logistique linéaire, table de décision, les coefficients Kappa et Matthews sont compris entre 0,6 et 0,7, ce qui correspond à de bonnes performances de classification. Les coefficients Kappa et Matthews les plus élevés, supérieurs à 0,7, sont obtenus pour les méthodes perceptron, C4.5, PART ou la forêt aléatoire. Parmi ces dernières, celles dont la durée d'apprentissage est faible sont conservées. Les trois méthodes ainsi sélectionnées sont la forêt aléatoire, le classifieur C4.5 et la table de décision de type PART.

Table 4.12: Comparaison des algorithmes appliqués à la base A2_modif. Les valeurs de Kappa et Matthews supérieures à 0,7 sont signalées par l'astérisque *.

	Taux de réussite	Kappa	Matthews	Apprentissage (sec)	Test (sec)
Réseau bayésien	95,67	0,53	0,57	2,07	0,02
Bayésien naïf	47,72	0,05	0,15	0,4	0,1
SVM	96,94	0	0	2773,42	69,39
Rég. log. mult.	98,06	0,64	0,64	7,22	0,01
Perceptron	98,56	0,75*	0,75*	404,66	0,06
Rég log. lin.	98,28	0,65	0,66	65,94	0,01
K-ppv	97,4	0,57	0,57	0,02	113,89
K*	97,39	0,52	0,52	0,01	2957,85
Table de décision	98,41	0,7*	0,7*	20,57	0,02
PART	98,73	0,78*	0,78*	4,56	0,01
C4.5	98,73	0,78*	0,78*	2,63	0,01
Forêt aléatoire	98,7	0,76*	0,76*	3,61	0,02

La base A8_modif présente globalement des temps d'apprentissage et de test plus

longs que la base A2_modif. Les performances des différents algorithmes sont très bonnes puisque les coefficients Kappa et Matthews ont des valeurs très proches de 1. Seules les méthodes SVM et K* affichent des performances plus faibles. Les différents indicateurs calculés pour cette base ne permettent pas de séparer les algorithmes de classification. La sélection obtenue précédemment est donc conservée pour étudier la base A8_modif.

TABLE 4.13: Compraison des algorithmes appliqués à la base A8_modif. Les valeurs de Kappa et Matthews supérieures à 0,7 sont signalées par l'astérisque *.

	Taux de réussite	Kappa	Matthews	Apprentissage (sec)	Test (sec)
Réseau bayésien	95,98	0,94*	1*	3,2	0,06
Bayésien naïf	94,1	0,91*	1*	0,26	0,22
SVM	51,62	0	0	6703,32	64,34
Rég. log. mult.	98,57	0,98*	1*	295,03	0,03
Perceptron	98,51	0,98*	1*	331,97	0,05
Rég log. lin.	98,57	0,98*	1*	196,68	0,02
K-ppv	97,31	0,96*	1*	0,01	57,25
K*	79,59	0,68	0,73*	0,01	1278,47
Table de décision	98,27	0,97*	1*	18,09	0,02
PART	98,79	0,98*	1*	4,9	0,01
C4.5	98,78	0,98*	1*	2,77	0,01
Forêt aléatoire	98,69	0,98*	1*	4,15	0,01

Optimisation des paramètres

Les algorithmes de classification sélectionnés lors de l'étape précédente sont la forêt aléatoire, C4.5 et PART. Cette partie concerne l'optimisation des paramètres pour chacune de ces méthodes sur les bases de données A2_modif et A8_modif.

L'algorithme de forêt aléatoire dépend de deux paramètres, le nombre I d'arbres à générer et le nombre K d'attributs à utiliser lors de la génération des arbres. D'après Bernard *et coll.* [BHA+08], la valeur la plus proche du K optimal est égale à la racine carrée du nombre d'attributs des données à étudier. Ici, l'arrondi à la valeur supérieure de cette valeur est pris en compte, K est donc égal à 6. La valeur de I est déterminée de manière empirique. Pour cela, différentes valeurs de I allant de 100 à 1200 par pas de 100 ont été utilisées comme paramètres de l'algorithme. Les coefficients Kappa et Matthews qui résultent de l'application des différentes forêts aléatoires aux bases de données A2_modif et A8_modif sont représentés en figure 4.6 et 4.7. Pour la base A2_modif, les performances des forêts d'arbres s'améliorent avec

l'augmentation du nombre d'arbres considérés. L'amélioration est cependant limitée puisque les indices Kappa et Matthews varient sur la quatrième décimale. Il faut également noter que le temps d'apprentissage augmente linéairement avec la taille des arbres. Afin de faire un bon compromis entre performance et temps de calcul, une forêt de 800 arbres est retenue pour la prochaine étape. En ce qui concerne la base A8_modif, quel que soit le nombre d'arbres considéré, les performances restent similaires. Le temps d'apprentissage augmente encore linéairement avec la taille des arbres. Pour ce cas, une forêt de 100 arbres sera considérée pour la prochaine étape.

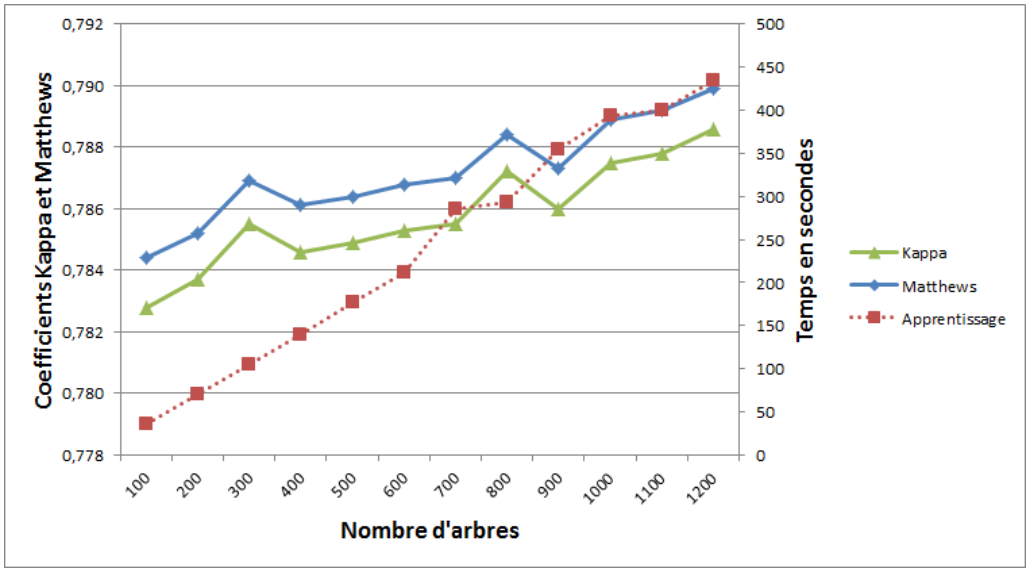


FIGURE 4.6 – Optimisation du paramètre nombre d'arbres des forêts aléatoires sur la base de données A2_modif

L'algorithme C4.5 dépend d'un indice de confiance C et de M, le nombre minimal d'instances par feuille qui déterminent la manière d'élaguer un arbre de décision. D'après Beck *et coll.* [BGZ+08], les paramètres par défaut pour C et M sont respectivement 0,25 et 2. Cette valeur de M est conservée car nous souhaitons couvrir le plus de cas possibles. Par contre, la valeur optimale de C est étudiée de manière empirique. Les différentes valeurs considérées sont comprises entre 0,05 et 1 par intervalle de 0,05. Les coefficients Kappa et Matthews résultant sont représentés en figure 4.8 pour la base de données A2_modif et 4.9 pour la base A8_modif. Comme pour les forêts d'arbres, les performances associées à la base A8_modif varient très peu avec le nombre de paramètres. Pour la base A2_modif, les variations des performances sont plus importantes. L'indice de confiance associé aux meilleures performances pour ces deux bases est 0,05.

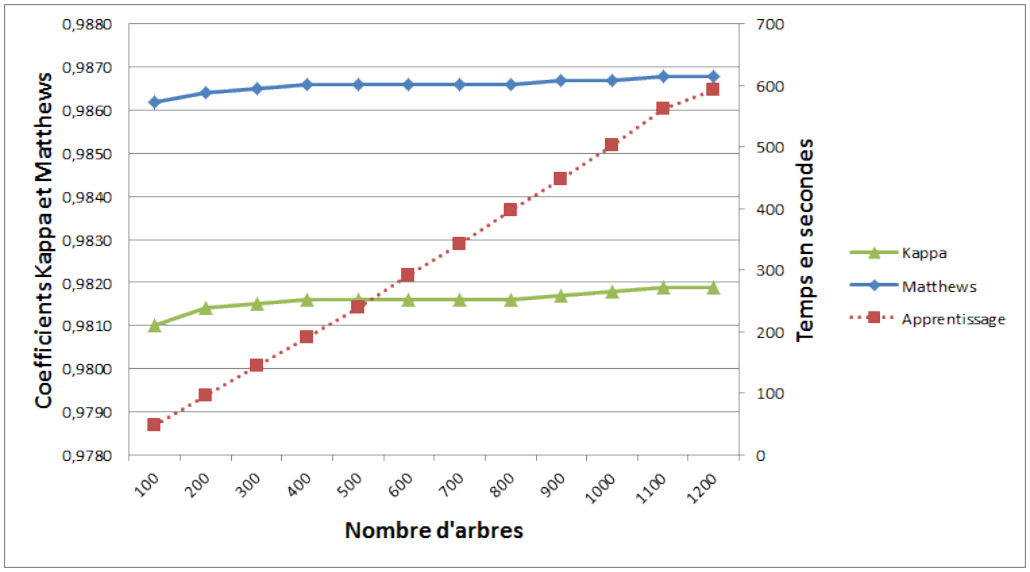


FIGURE 4.7 – Optimisation du paramètre nombre d'arbres des forêts aléatoires sur la base de données A8_modif

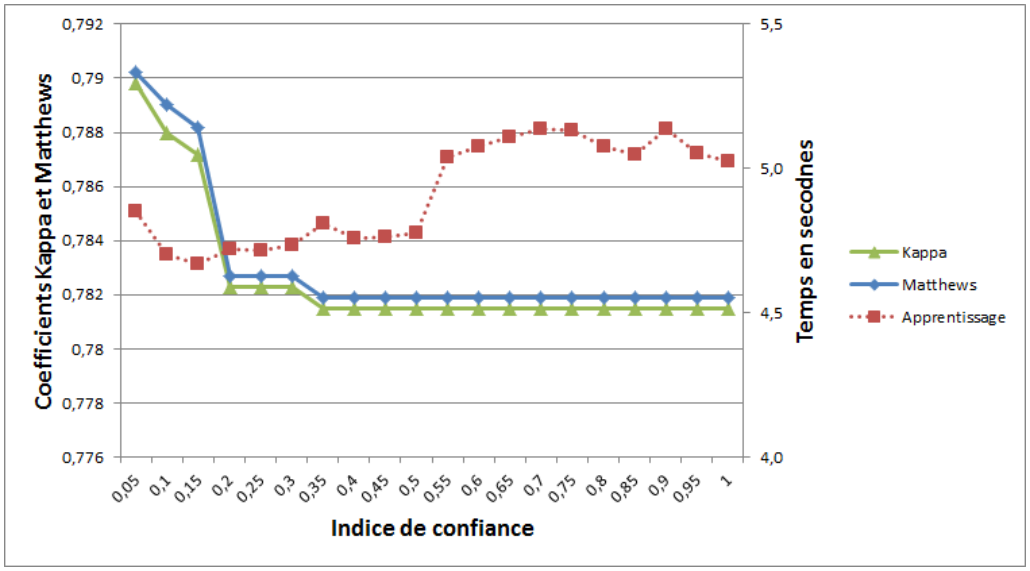


FIGURE 4.8 – Optimisation du paramètre indice de confiance du classifieur C4.5 sur la base de données A2_modif

La table de décision PART est construite en construisant plusieurs arbres de décision C4.5, puis en retenant les feuilles les plus représentatives pour les transformer en règles. PART dépend donc des mêmes paramètres C et M que l'algorithme précédent C4.5. La même méthodologie que pour l'optimisation des paramètres de C4.5 est appliquée ici. Les mesures Kappa et Matthews obtenues sont représentées en figure 4.10 pour la base A2_modif et 4.11 pour la base A8_modif. Comme précédemment,

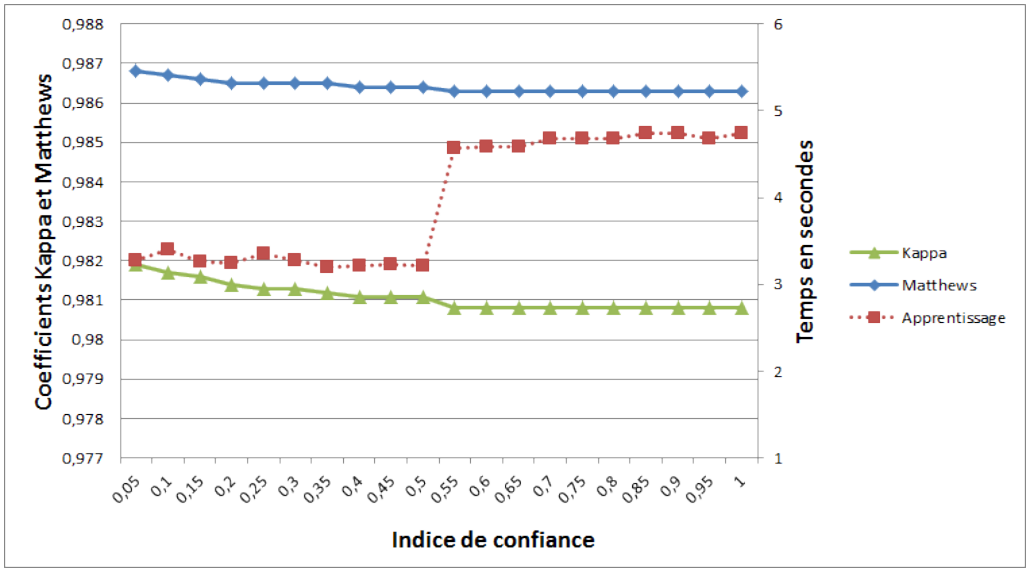


FIGURE 4.9 – Optimisation du paramètre indice de confiance du classifieur C4.5 sur la base de données A8_modif

les meilleurs résultats sont obtenus pour $C = 0,05$. On peut également remarquer que le temps d'apprentissage varie de manière similaire que celui de l'algorithme C4.5.

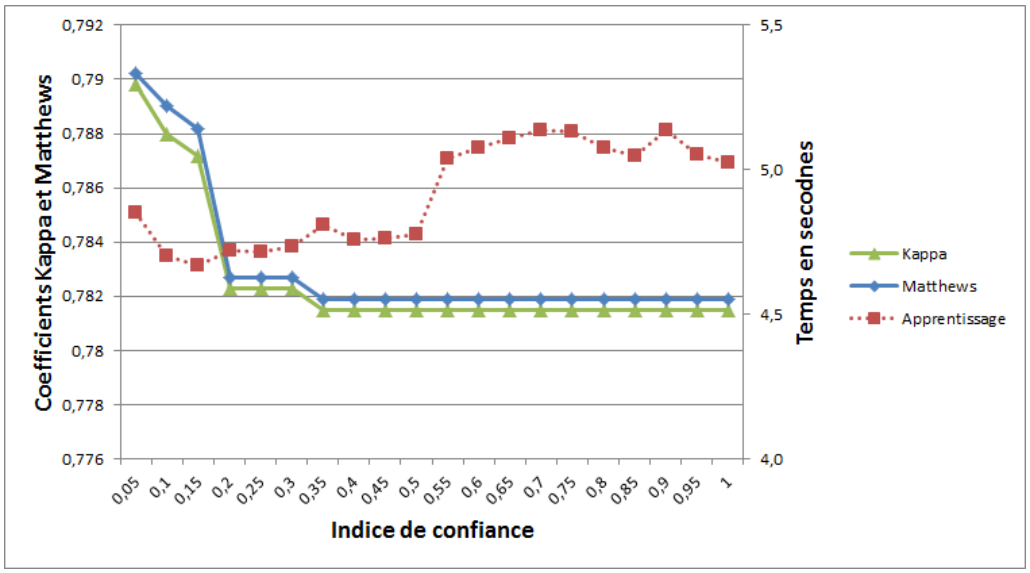


FIGURE 4.10 – Optimisation du paramètre indice de confiance de la table de décision de type PART sur la base de données A2_modif

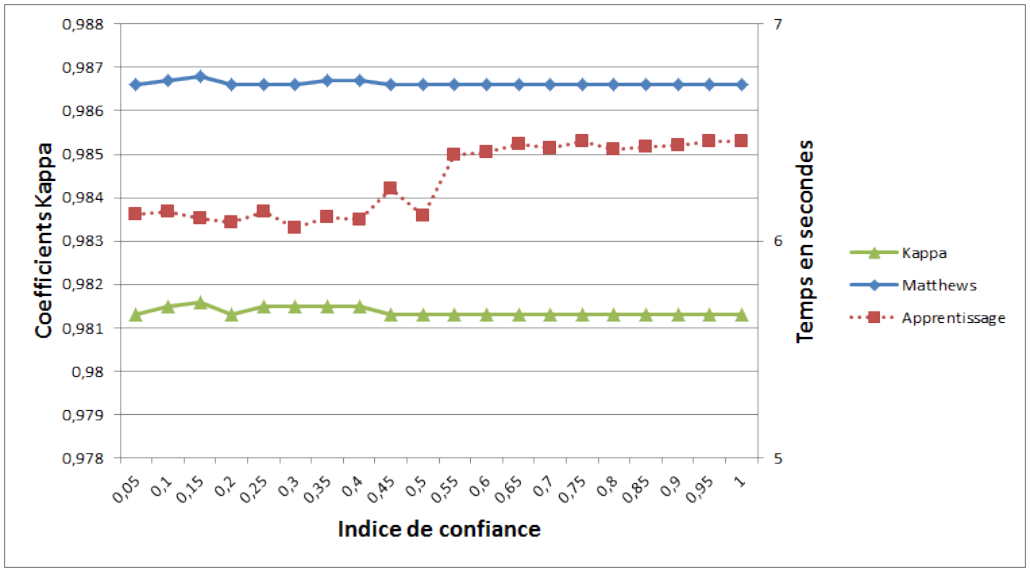


FIGURE 4.11 – Optimisation du paramètre indice de confiance de la table de décision de type PART sur la base de données A8_modif

Validation

Les étapes précédentes ont conduit au choix de trois méthodes de classification et à l'optimisation de leurs paramètres. Afin de valider ces choix, ces algorithmes et leurs paramètres optimaux sont utilisés pour classifier les données correspondant à la base de données B. Les bases A2_modif et A8_modif servent de base d'apprentissage. Les deux modèles qui résultent de ces apprentissages sont ensuite utilisés pour classifier respectivement les données des bases B2_modif et B8_modif. L'objectif est de vérifier si les algorithmes sélectionnés permettent de classifier les données d'une base qui suit une distribution statistique similaire.

Le tableau 4.14 regroupe les performances des algorithmes optimisés appliqués aux jeux de données dont la vérité terrain est constituée de deux classes A2_modif et B2_modif. Les coefficients Kappa et Matthews sont inférieurs à ceux observés avec les mêmes paramètres sur la base A2_modif seule mais restent supérieurs à 0,7. Les temps d'apprentissage et de test restent similaires. Ces résultats montrent que les algorithmes sélectionnés et leurs paramètres s'adaptent bien à une autre base de données qui respecte une distribution statistique similaire à celle utilisée pour réaliser l'apprentissage.

Les matrices de confusion 4.15, 4.16 et 4.17 montrent que leurs coefficients ont des ordres de grandeur similaires. Cependant, le classifieur C4.5 est celui qui présente

TABLE 4.14: Validation des choix pour les bases à 2 classes.

	Forêt aléatoire	C4.5	PART
	I=800	C=0,05	C=0,05
Kappa	0,7445	0,7398	0,7357
Matthews	0,745	0,740	0,736
Temps d'apprentissage (s)	2,67	2,65	4,64
Temps de test (s)	21,38	0,7	0,78

le plus de transactions frauduleuses et normales mal classifiées. La table de décision PART induit le plus petit nombre de fraudes mal classifiées mais également le plus fort nombre de transactions normales détectées comme frauduleuses. C'est quasiment le contraire pour la forêt aléatoire qui présente un très fort taux de classifications frauduleuses mal classifiées et le plus faible nombre de transactions légitimes mal détectées.

TABLE 4.15: Matrice de confusion liée à la forêt aléatoire de 800 arbres appliquée à la base A2_modif

		Prédit	
		Légitime	Fraude
Réel	Légitime	52 187	345
	Fraude	464	1 228

TABLE 4.16: Matrice de confusion liée au classifieur C4.5 d'indice de confiance 0,05 appliquée à la base A2_modif

		Prédit	
		Légitime	Fraude
Réel	Légitime	52 177	355
	Fraude	470	1 222

TABLE 4.17: Matrice de confusion liée à la table de décision PART d'indice de confiance 0,05 appliquée à la matrice A2_modif

		Prédit	
		Légitime	Fraude
Réel	Légitime	52 105	427
	Fraude	437	1 255

Les performances des algorithmes optimisés appliqués aux jeux de données A8_modif et B8_modif sont regroupées dans le tableau 4.18. Comme précédemment, les coefficients Kappa et Matthews sont légèrement inférieurs à ceux observés

avec les mêmes paramètres lors de la phase d'optimisation. La stabilité des algorithmes montre qu'il n'y a pas eu de sur- ou sous-apprentissage.

TABLE 4.18: Validation des choix pour les bases à 8 classes

	Forêt aléatoire	C4.5	PART
	I=100	C=0,05	C=0,05
Kappa	0,9768	0,9774	0,977
Matthews	0,983	0,984	0,983
Temps d'apprentissage (s)	112,13	3,62	6,42
Temps de test (s)	6,93	0,8	0,81

Les tables 4.19, 4.20 et4.21 correspondent aux matrices de confusion des différentes méthodes de classification appliquées aux jeux de données A8_modif pour l'apprentissage et B8_modif pour la phase de test.

TABLE 4.19: Matrice de confusion liée à la forêt aléatoire de 800 arbres appliquée aux bases A8_modif et B8_modif

		Prédit							
		a	b	c	d	e	f	g	h
Réel	a	**12 784**	0	0	0	0	0	0	0
	b	0	**27 901**	0	0	0	0	0	0
	c	0	0	**7 166**	0	0	0	**338**	0
	d	0	0	0	**3 886**	0	**12**	0	**1**
	e	0	0	0	0	**442**	0	0	0
	f	0	0	0	**59**	0	**173**	0	0
	g	0	0	**394**	0	0	0	**337**	0
	h	0	0	0	**14**	0	**4**	0	**711**

a=N-RegDep, b=N-RegRC, c=N-RegC2C, d=N-RegWith, e=N-RegMerch, f=F-SevWith, g=F-bot, h=F-Mule-With

Les trois matrices montrent également que les transactions légitimes liées aux habitudes de type dépôt, achat de temps de communication, paiement marchand sont toujours classifiées correctement, ce qui n'est pas le cas des transactions légitimes liées aux habitudes de type transfert entre particuliers et retrait. Cette observation peut s'expliquer par le fait que toutes les fraudes correspondent à ces deux derniers types de transactions.

De manière générale, les classes qui sont confondues sont celles dont le type de transaction est le même. Ainsi, pour le classifieur C4.5, 15 retraits réalisés par les

TABLE 4.20: Matrice de confusion liée au classifieur C45 d'indice de confiance 0,05 appliquée aux bases A8_modif et B8_modif

		Prédit							
		a	b	c	d	e	f	g	h
Réel	a	**12 784**	0	0	0	0	0	0	0
	b	0	**27 901**	0	0	0	0	0	0
	c	0	0	**7 180**	0	0	0	**324**	0
	d	0	0	0	**3 875**	0	**18**	0	**6**
	e	0	0	0	0	**442**	0	0	0
	f	0	0	0	**57**	0	**173**	0	**2**
	g	0	0	**373**	0	0	0	**358**	0
	h	0	0	0	**15**	0	**8**	0	**706**

a=N-RegDep, b=N-RegRC, c=N-RegC2C, d=N-RegWith, e=N-RegMerch, f=F-SevWith, g=F-bot, h=F-Mule-With

TABLE 4.21: Matrice de confusion liée à la table de décision PART d'indice de confiance 0,05 appliquée aux bases A8_modif et B8_modif

		Prédit							
		a	b	c	d	e	f	g	h
Réel	a	**12 784**	0	0	0	0	0	0	0
	b	0	**27 901**	0	0	0	0	0	0
	c	0	0	**7 194**	0	0	0	**310**	0
	d	0	0	0	**3 873**	0	**18**	0	**8**
	e	0	0	0	0	**442**	0	0	0
	f	0	0	0	**58**	0	**174**	0	0
	g	0	0	**402**	0	0	0	**329**	0
	h	0	0	0	**18**	0	**4**	0	**707**

a=N-RegDep, b=N-RegRC, c=N-RegC2C, d=N-RegWith, e=N-RegMerch, f=F-SevWith, g=F-bot, h=F-Mule-With

mules sont détectés en tant que retraits légitimes et 8 autres sont détectés en tant que retraits par un voleur. Le nombre de mauvaises classifications le plus important est rencontré pour les transferts entre particuliers légitimes et les transferts réalisés par les terminaux infectés. Ces deux types de transactions sont confondues par les algorithmes. Le fait que la fréquence des transferts frauduleux effectués par un terminal infecté est plus faible que celle des retraits effectués par les voleurs peut expliquer cette différence puisque les premiers sont plus proches de transactions légitimes que les seconds.

Le classifieur C4.5 est associé à 445 transactions frauduleuses détectées en tant que

transactions légitimes et 348 transactions légitimes détectées en tant que légitimes. La table de décision de type PART résulte en 478 transactions frauduleuses mal classifiées et 336 transactions légitimes mal classifiées. La forêt aléatoire donne 467 fraudes mal classifiées et 351 fausses transactions légitimes. La méthode qui implique le moins de fausses transactions légitimes est le classifieur C4.5 tandis que la table de décision PART est celle qui implique le moins de transactions légitimes mal classifiées.

4.2.3 Discussion sur l'adaptation des algorithmes de classification

Dans cette partie, l'adaptation d'algorithmes de classification a été étudiée en quatre étapes. Tout d'abord, différentes représentations de données ont été examinées. Parmi les différentes représentations, deux ont été sélectionnées pour les trois étapes suivantes. Ensuite, douze algorithmes de classification paramétrés par défaut ont été appliqués à ces deux jeux de données. Ceux qui présentent les meilleurs résultats, la forêt aléatoire, la table de décision PART et le classifieur C4.5, ont été retenus pour ensuite optimiser leurs paramètres. Finalement, les algorithmes et leurs paramètres optimaux sélectionnés sont utilisés pour classifier les données d'une base qui possède des caractéristiques statistiques semblables à celles de la base utilisée pour l'apprentissage de l'algorithme. Cette dernière étape permet de confirmer les choix réalisés précédemment.

Les représentations de données qui affichent les meilleurs résultats sont celles pour lesquelles les différentes habitudes des utilisateurs sont spécifiées dans la vérité terrain. Les jeux de données pour lesquels différentes agrégations ont été calculées présentent également de bons résultats. Ces deux observations s'expliquent par le niveau de détail de ces deux types de représentations. En effet, la division des transactions légitimes en plusieurs catégories crée des classes disjointes et facilite ainsi la tâche des algorithmes. Quant à l'agrégation de certaines informations sur une heure, une journée ou une semaine, elle met en évidence des fréquences ou des modifications de comportement ce qui facilite la classification.

A notre connaissance, les algorithmes à base de règles sont les plus utilisés par les banques parce qu'ils facilitent l'interprétation des résultats. D'après l'étude menée dans cette thèse, ces algorithmes sont également les plus performants pour le problème considéré ici. En particulier, les forêts aléatoires sont les méthodes les plus performantes en terme de taux de classification. Al-Khatib [AK12] identifie et liste

les études comparatives d'algorithmes sur le problème de la détection de fraude sur les transactions réalisées par carte bancaire. Les forêts aléatoires n'apparaissent pas dans les différentes études citées.

En revanche, deux études citées mettent en oeuvre certains algorithmes utilisés ici. Gadi *et coll.* [GWdL08] compare cinq types de méthodes, un réseau de neurones, un réseau bayésien, un réseau bayésien naïf et les arbres de décision. Les deux méthodes les plus performantes dans cette étude sont le réseau de neurones et le réseau bayésien. Abott *et coll.* [AMEI98], quant à eux comparent différents outils qui implémentent des arbres de décision, des réseaux de neurones, des régressions, des méthodes considérant les plus proches voisins. D'après cette étude, les méthodes à base d'arbres de décision et de réseaux de neurones sont les plus performantes. Les résultats de ces études sont difficilement comparables à ceux obtenus ici. En effet, ces comparaisons ont été réalisées avec des bases de données différentes et des méthodologies différentes.

Les résultats obtenus ici peuvent être complétés en étudiant d'autres représentations de données. Par exemple, les données pourraient être normalisées. Ainsi, les différents attributs des données auraient le même ordre de grandeur et donc le même poids dans divers calculs comme les distances. D'autres types de données pourraient également être incluses comme le lieu de la transaction qui n'est pas pris en compte dans le simulateur.

Les coûts de traitement des classifications pourraient également être pris en compte. En effet, la classification d'une fraude en tant que transaction légitime entraîne une perte financière et d'image pour l'établissement alors que la classification d'une transaction légitime en tant que fraude peut entraîner des désagréments pour les utilisateurs et des coûts de vérification non nécessaires. Il est généralement considéré que ces deux types de coût n'ont pas le même poids et cela peut entraîner des performances différentes que celles énoncées ici. Il appartient aux établissements de définir ces différents coûts en fonction de leur politique de gestion des risques. Gadi *et coll.* [GWdL08] proposent quant à eux qu'une transaction frauduleuse mal classifiée soit associée à un coût de 100$, qu'une transaction légitime mal classifiée soit associée à un coût de 10$ et que l'analyse d'une transaction réellement frauduleuse coûte 1$.

À titre d'exemple, les coûts des algorithmes PART, C4.5 et forêt aléatoire munis de leurs paramètres optimaux pour les bases à deux classes sont respectivement de

49 225$, 51 772$ et 51 078$. Les coûts de ces mêmes algorithmes et leurs paramètres optimaux pour les bases à huit classes sont respectivement de 52 374$, 49 227$ et 51 435$. En considérant le coût, l'algorithme PART semble être le plus intéressant pour les bases où la vérité terrain est décrite par deux classes. De même l'algorithme C4.5 apparaît comme le plus intéressant pour les bases A8_modif et B8_modif.

Le coût pourrait également être utilisé comme méthode d'évaluation pour diriger la phase d'apprentissage. Cette étude pourrait être poursuivie en considérant cette option.

Une autre perspective de ces travaux serait d'étudier les transactions mal classifiées. Les questions qui pourraient être ainsi abordées sont : est ce que ce sont les mêmes transactions qui ne sont pas détectées ? Y a-t-il des transactions mal classifiées pour un seul algorithme ? Quelles sont les caractéristiques de ces transactions ?

Finalement, ces travaux ont été réalisés sur une base de données synthétiques. Les résultats obtenus peuvent être influencés par des biais de ces données. L'étude pourrait également être élargie à l'application des algorithmes sur des données réelles labellisées.

4.3 Discussion

Ce chapitre détaille les contributions de cette thèse dans le domaine de la détection de fraudes pour les services de transactions sur terminaux mobile.

Un simulateur de données qui modélise le fonctionnement de ces services et du comportement de ses utilisateurs a été réalisé. Une validation préliminaire a également été effectuée. Bien que l'utilisation de données peut prêter à discusion, notemment parce qu'étudier un algorithme avec ces données ne garantit pas ses capacités sur des données réelles, les données synthétiques présentent de nombreux avantages. Elles permettent entre autres de s'affranchir des problématiques de gestion de la vie privée et de la quantité de données disponibles.

De nombreuses perspectives existent autour de ce simulateur. La modélisation des utilisateurs pourrait être approfondie en affinant la simulation des porteurs en introduisant les comportements des marchands et des vendeurs de monnaie électronique et en étoffant les fraudes modélisées. La validation pourrait également être poursuivie

en étudiant la possibilité d'avoir plusieurs habitudes par type de transactions. Enfin des simulations impliquant plus d'utilisateurs ou des comportements plus complexes pourraient être créées.

Une autre perspective des travaux concernant le simulateur pourrait être de créer plusieurs parcours fraudeurs et d'étudier la réaction des algorithmes de détection face à ces différentes stratégies.

Ce simulateur a permis de générer plusieurs jeux de données qui ont été utilisées pour réaliser des démonstrateurs et pour développer des outils dans le cadre du projet européen MASSIF. Des jeux de données synthétiques ont également été générés pour la deuxième contribution de cette thèse.

Différentes représentations de données ont été étudiées. Ensuite, douze algorithmes ont été comparés en trois étapes au cours desquelles les moins performants ont été éliminés. Au final, trois méthodes et leurs paramètres optimisés ont été sélectionnés et testés sur une base jamais utilisée dans les trois premières étapes mais dont les caractéristiques statistiques sont semblables à la base d'apprentissage. Les bons résultats obtenus lors de cette phase de validation montrent que les méthodes et paramètres sélectionnés ne donnent pas lieu à sur- ou sous-apprentissage.

Une première interprétation des résultats des classifications a été réalisée. Nous mettons ainsi en évidence que la précision des classes indiquées dans la vérité terrain permet d'améliorer les performances des algorithmes. Nous montrons également que les données agrégées sur des périodes de temps permettent d'obtenir de meilleurs résultats. Enfin, les algorithmes prennent bien en compte le fait que les fraudes sont réalisées sur quelques types de transactions seulement. Les trois algorithmes qui resortent sont PART, C4.5 et la forêt aléatoire.

Cette contribution implique également plusieurs perspectives. En particulier, l'étude pourrait être étendue à d'autres représentations de données et à d'autres méthodes. Il serait également nécessaire de confronter les observations réalisées avec des algorithmes appliqués sur des données réelles labellisées.

Conclusions et perspectives

Bilan

La problématique de cette thèse concerne la sécurisation des systèmes de transactions sur terminaux mobiles. Les transactions sont réalisées à l'aide de monnaie électronique privative gérée par l'opérateur de télécommunication qui fournit le service. Cette thèse se place dans le cadre où le système se base sur l'utilisation de réseaux tout-IP et de terminaux mobiles comme des téléphones intelligents, *smartphones*. Le système considéré se base sur un modèle paiement trois coins, ce qui implique qu'il s'agit d'un circuit fermé auquel appartient tous les acteurs.

La sécurisation de ce service est abordée sous deux angles complémentaires. Dans un premier temps, la question de l'architecture de sécurité a été traitée. L'objectif est de proposer une architecture et des protocoles qui sont adaptés aux réseaux tout-IP, qui tirent profit des capacités des fonctionnalités de ces réseaux et des smartphones pour proposer des services sécurisés de bout-en-bout et plus ergonomiques. Dans un second temps, la fraude et la détection de la fraude dans ce domaine ont été étudiées. L'objectif est d'adapter des méthodes de classification à ces systèmes.

Bien que des travaux dans le domaine de la fraude bancaire existent, aucune étude ne s'est penchée sur la détection de fraude dans les services de transaction sur mobile et l'adaptation des méthodes de classification à ce domaine. La différence des usages et des modèles, par rapport aux transactions bancaires, implique qu'il est nécessaire de réaliser une adaptation propre au service de paiement sur terminal mobile.

Le premier axe de recherche sur lequel nous avons travaillé concerne la sécurité et les moyens permettant d'éviter que des fraudes se réalisent en exploitant l'architecture

réseau de ce système. Nous avons proposé une architecture et des protocoles qui permettent de sécuriser la transaction de bout-en-bout entre une carte SIM d'un terminal mobile et la plateforme de paiement. Cette architecture est adaptée aux réseaux de type tout-IP comme les réseaux 4G. Nous avons associé les fonctionnalités de sécurité de la carte SIM à celles des environnements d'exécution sécurisée dans les terminaux pour réduire l'exploitation des risques que nous avons identifiés.

Différents modes de transactions ont été proposés. Ils permettent de réaliser des transferts entre particuliers ainsi que des paiements marchands. Ceux-ci peuvent être effectués en mode connecté, si le porteur, le marchand et la plateforme sont en ligne et peuvent dialoguer entre eux. Nous proposons également un mode semi-connecté qui permet aux porteurs n'ayant pas de forfait d'accès aux données de réaliser des paiements. Pour cela, nous suggérons que le marchand partage sa connexion au réseau avec les porteurs pour qu'ils puissent ensuite réaliser la procédure de paiement connectée. Finalement, nous proposons un mode déconnecté pour des cas dégradés où les deux acteurs de la transaction n'ont temporairement pas accès au réseau pour dialoguer avec la plateforme de paiement.

Les différents protocoles proposés ont été validés au niveau du respect des objectifs de sécurité ainsi que des performances et des usages. Une vérification formelle a été réalisée à l'aide d'un outil qui implémente différentes méthodes automatiques de vérification formelle. Cette analyse a identifié une attaque contre le paiement marchand. Cependant, celle-ci parait peu réalisable étant donné que chaque partie de la transaction, marchand ou porteur, reçoit directement de la plateforme de paiement des informations récapitulatives de la transaction. De plus, des implémentations particulières à base de protocoles délimiteurs de distance peuvent permettre d'éviter cette attaque. Nous avons évalué les temps de traitement par la carte SIM pour chacun des protocoles. Certains résultats ne semblent pas bons mais peuvent être améliorés avec des cartes plus performantes que celles utilisées dans le cadre de l'étude.

Le second axe de recherche abordé dans cette thèse concerne l'adaptation des méthodes de classification au problème de détection de fraudes dans les systèmes de transaction sur terminaux mobiles. Cette étude se concentre sur les fraudes de type comportemental pour lesquelles le comportement du fraudeur et de sa victime se superposent. Ce type de fraude ne dépendent pas de la typologie des réseaux et, selon les scénarios considérés, peuvent prendre place dans les systèmes de transaction sur terminaux mobiles actuels. Afin de traiter cette problématique, nous

avons tout d'abord proposé un outil qui génère des données artificielles. L'objectif était de résoudre notre problème lié au manque de données labellisées et contenant des fraudes. Cet outil simule le fonctionnement de la plateforme de paiement, en particulier les fonctions d'authentification et d'autorisation de la transaction, ainsi que le comportement des utilisateurs. La modélisation des comportements est basée sur l'hypothèse que ceux-ci sont constitués d'habitudes.

Une vérification préliminaire a été réalisée à partir de données provenant d'un système réel. Cette base de données nous a également permis de paramétrer l'outil afin d'obtenir une modélisation complexe dans laquelle des simulations de fraudes ont été introduites. Les données synthétiques ainsi obtenues ont été utilisées pour notre étude sur l'adaptation des méthodes de classification au problème ciblé par la thèse.

Cette seconde partie a été réalisée en plusieurs étapes. Tout d'abord, une représentation des données associée à de meilleurs résultats a été sélectionnée. Ensuite, douze algorithmes de classification ont été évalués avec leurs paramètres par défaut. Cette phase nous a permis de retenir trois algorithmes pour optimiser leurs paramètres. A la suite de cette optimisation, une forêt aléatoire de 800 arbres, un classifieur C45 avec un indice de confiance de 0,05 et une table de décision PART avec un indice de confiance de type 0,05 ont été choisis. Finalement, ce choix a été validé en appliquant ces algorithmes pour classifier des données d'une base de données différente de celle utilisée pour l'étude. Une première interprétation de ces résultats a été réalisée. Nous mettons ainsi en évidence que la précision des classes indiquées dans la vérité terrain permet d'améliorer les performances des algorithmes. Nous montrons également que les données agrégées sur des périodes de temps permettent d'obtenir de meilleurs résultats. Enfin, les algorithmes prennent bien en compte le fait que les fraudes sont réalisées sur quelques types de transactions seulement.

Contributions

Les différentes contributions réalisées au cours de cette thèse sont :

- une architecture de sécurité et des protocoles de paiement basés sur l'utilisation d'un réseau tout-IP et de terminaux mobiles comme les *smartphones* ;
- des protocoles de paiement qui assurent la sécurité de bout-en-bout au niveau applicatif entre un élément sécurisé du terminal mobile et la plateforme de paiement ;

- une validation des protocoles de sécurité proposés. Tout d'abord, la satisfaction des besoins de sécurité revendiqués est vérifiée formellement à l'aide de l'outil AVISPA [AVI03b]. Ensuite, les performances attendues des protocoles sont également pris en compte ;
- un simulateur permettant de générer des données de transactions synthétiques comprenant des comportements légitimes et frauduleux modélisés ;
- l'exploitation de données d'un système de transactions sur mobile déployé sur le terrain pour créer et paramétrer le simulateur ;
- une étude préliminaire sur l'adaptation d'algorithmes de classification au domaine de la détection de la fraude dans les systèmes de transaction sur terminaux mobiles.

Perspectives

Les travaux de cette thèse présentent plusieurs perspectives. Tout d'abord, l'architecture et les protocoles proposés pourraient être implémentés et évalués.

Le simulateur de données pourrait être amélioré en incluant le comportement des marchands et des agents distribuant la monnaie électronique. Les parcours fraudeurs modélisés pourraient être développés et complexifiés. La modélisation des comportements des utilisateurs pourrait également être affinée en considérant la possibilité d'avoir plusieurs habitudes par type de transaction. Cette notion pourrait également être utilisée pour continuer la validation de l'hypothèse qui sous-tend le simulateur. Cette perspective est particulièrement importante car l'accès à une base de données réelle, correctement labellisée et pouvant être partagée est très difficile.

Des jeux de données publics pourraient être créés à l'aide de ce simulateur de données. Ces jeux pourraient être utilisés pour réaliser des études comparatives sur les algorithmes et outils de détection de fraude. Cette perspective est très intéressante puisqu'aujourd'hui de tels travaux ne peuvent être réalisés du fait du manque de données exploitables.

Au-delà de l'utilisation de jeux de données, le simulateur peut être utilisé pour tester des outils de gestion de la fraude. Par exemple, dans le cadre du projet euréopen FP7 MASSIF, le simulateur a été exploité pour évaluer les résultats du projet [Masre]. Il a également été utilisé comme base de deux démonstrateurs qui ont été utilisés lors de l'évaluation du projet par la Commission Européenne. Le second démonstrateur a

également présenté lors du workshop RaSIEM [1] qui présente les résultats du projet européen FP7 MASSIF. Dans ces deux démonstrateurs, le simulateur est associé aux outils de détection et d'application de contre-mesures développés dans le cadre du projet. Il a permis de montrer en temps réel le fonctionnement de ces outils et de leur impact sur la plateforme de paiement et les actions réalisées par les utilisateurs.

L'étude sur l'adaptation des algorithmes de classification pourrait être complétée en considérant plus de représentations de données. Par exemple, des données normalisées ou contenant des informations supplémentaires comme la localisation pourraient également être prises en compte. Cette étude pourrait également être complétée en considérant plus d'algorithmes de classification ou d'algorithmes d'autres catégories. Des données contenant plus de transactions et d'acteurs pourraient être utilisées. De plus, les habitudes des utilisateurs pourraient évoluer au cours du temps. L'aspect temporel des transactions pourrait également être pris en compte. Enfin, il serait intéressant d'obtenir des transactions réelles associées à une vérité terrain afin de les confronter aux observations réalisées ici.

1. http://www.ares-conference.eu/conf/index.php?option=com_content&view=article&id=78&Itemid=107

Publications de l'auteur

Conférences internationales avec comité de lecture et avec actes

1. **Chrystel GABER**, Baptiste HEMERY, Mohammed ACHEMLAL, Marc PASQUET et Pascal URIEN. « Synthetic logs generator for fraud detection in mobile transfer services ». In : The International Symposium on Collaborative Technologies and Systems (CTS). 2013.
2. Roland RIEKE, Maria ZHDANOVA, Juergen REPP, Romain GIOT et **Chrystel GABER**. « Fraud Detection in Mobile Payments Utilizing Process Behavior Analysis ». In : International Conference od Availability, Reliability and Security (ARES). 2013.
3. Gustavo Gonzalez GRANADILLO, Herve DEBAR, Gregoire JACOB, **Chrystel GABER** et Mohammed ACHEMLAL. « Individual Countermeasure Selection based on the Return On Response Investment Index ». In : Mathematical Methods, Models, and Architectures for Computer Network Security. T 7531. Lecture Notes in Computer Science. 2012
4. Johann VINCENT, Vincent ALIMI, Aude PLATEAUX, **Chrystel GABER** et Marc PASQUET. « A Mobile Payment Evaluation Based on a Digital Identity Representation ». In : IEEE International Conference on Collaboration Technologies and Systems (CTS). 2012.
5. **Chrystel GABER**, Saïd GHAROUT, Mohammed ACHEMLAL, Marc PASQUET et Pascal URIEN. « Fraud detection in mobile payments ». In : International Conference on Secure Networking and Applications (ICSNA). 2011.
6. Mohamed ACHEMLAL, Jean-Claude PAILLÈS et **Chrystel GABER**. « Building Trust in Virtualized Networks ». In : The Second International Conference on Evolving Internet INTERNET 2010.

7. Jean-Claude PAILLÈS, **Chrystel GABER**, Vincent ALIMI et Marc PASQUET. « Payment and privacy : a key for the development of nfc mobile ». In : The 2010 International Symposium on Collaborative Technologies and Systems (CTS). 2010.

Conférences nationales avec comité de lecture et avec actes

1. **Chrystel GABER**, Mohammed ACHEMLAL, Baptiste HEMERY, Marc PASQUET et Pascal URIEN. « Réseaux 4G : anticiper la sécurité des systèmes de transactions sur mobile ». In : Conférence sur la Sécurité des Architectures Réseaux et des Systèmes d'Information (SARSSI). 2013.
2. **Chrystel GABER**, Saïd GHAROUT, Mohammed ACHEMLAL, Marc PASQUET et Pascal URIEN. « Security challenges of mobile money transfer services ». In : Conférence sur la Sécurité des Architectures Réseaux et des Systèmes d'Information (SARSSI). 2012.
3. **Chrystel GABER**, Romain GIOT, Mohammed ACHEMLAL, Baptiste HEMERY, Marc PASQUET et Pascal URIEN. « Analyse des comportements dans un système de transactions sur terminal mobile ». In : Conférence sur la sécurité des architectures réseaux et des systèmes d'information (SARSSI). 2012.
4. Mohamad ACHEMLAL, Saïd GHAROUT et **Chrystel GABER**. « Trusted Platform Module as an Enabler for Security in Cloud Computing ». In : Conférence sur la sécurité des architectures réseaux et des systèmes d'information (SARSSI). 2011
5. **Chrystel GABER** et Jean-Claude PAILLÈS. « Security and trust for mobile phones based on virtualization ». In : The third Norsk Information security conference (NISK). 2010.

Rapports de recherche

1. MASSIF. D2.3.2 Deployment. Rapp. tech. MASSIF FP7-257475, 2013 - à publier en Septembre.
2. MASSIF. D2.3.3 Test and Evaluation. Rapp. tech. MASSIF FP7-257475, 2013 - à publier en Septembre.

3. **Chrystel GABER**. Bilan de deuxième année, Ecole Doctorale SIMEM. Rapp. tech. Ecole Doctorale SIMEM, 2012.
4. **Chrystel GABER**. Bilan de deuxième année de thèse, Orange Labs. Rapp. tech. Orange Labs, 2012.
5. **Chrystel GABER**. Etat de l'art des méthodes de fouilles de données pouvant être utilisées pour la détection de fraudes. Rapp. tech. Orange Labs, 2012.
6. MASSIF. D6.1.2 MASSIF Workshop Report. Rapp. tech. MASSIF FP7-257475, 2012.
7. **Chrystel GABER**. Bilan de première année de thèse, Ecole Docotorale SIMEM. Rapp. tech. Ecole Doctorale SIMEM, 2011.
8. **Chrystel GABER**. Bilan de première année de thèse, Orange Labs. Rapp. tech. Orange, 2011.
9. **Chrystel GABER**. D2.3.1 Evaluation Plan. Rapp. tech. MASSIF FP7-257475, 2011.
10. MASSIF. D6.2.1 Exploitation plan. Rapp. tech. MASSIF FP7-257475, 2011.
11. MASSIF. D2.1.1 Scenario requirements. Rapp. tech. MASSIF FP7-257475, 2011.

Poster

1. **Chrystel GABER**, Baptiste HEMERY, Mohamed ACHEMLAL, Marc PASQUET et Pascal URIEN. « Synthetic logs generator for fraud detection in mobile transfer services ». In : Financial Cryptography and Data Security. 2013.

Brevet

1. Mohammed ACHEMLAL et **Chrystel GABER**. « Mode de sécurisation des transactions sur mobiles s'appuyant sur un "Secure Element" ». Brev. 08559. 2013.

Bibliographie

[3GP09] 3GPP : Dispelling lte myths. http://www.3gpp.org/Dispelling-LTE-Myths, November 2009. Dernière visite le 24/08/2013. [cité p. 21]

[AC04] Alessandro Armando et Luca Compagna : Satmc : A sat-based model checker for security protocols. *In Logics in Artificial Intelligence*, pages 730–733. Springer, 2004. [cité p. 74]

[AGG+11] Mohammed Achemlal, Saïd Gharout, Chrystel Gaber, Marc Llanes, Elsa Prieto, Rodrigo Diaz, Luigi Coppolino, Antonio Sergio, Rosario Cristaldi, Andrew Hutchison et Keiran Dennie : Scenario requirements. Rapport technique, MASSIF FP7-257475, 2011. [cité p. 92, 95, 96]

[AK12] Adnan Al-Khatib : Electronic payment fraud detection techniques. *World of Computer Science and Information Technology Journal (WCSIT)*, 2:137–141, 2012. [cité p. 48, 128]

[AK13] Gildas Avoine et Chong Hee Kim : Mutual distance bounding protocols. *IEEE Transactions on Mobile Computing*, 12(5):830–839, 2013. [cité p. 83, 88]

[AKA91] David W Aha, Dennis Kibler et Marc K Albert : Instance-based learning algorithms. *Machine learning*, 6(1):37–66, 1991. [cité p. 46, 116]

[AKM+05] William Aiello, Charles Kalmanek, Patrick McDaniel, Subhabrata Sen, Oliver Spatscheck et Jacobus Merwe : Analysis of communities of interest in data networks. *In* Constantinos Dovrolis, éditeur : *Passive and Active Network Measurement*, volume 3431 de *Lecture Notes in Computer Science*, pages 83–96. Springer Berlin Heidelberg, 2005. [cité p. 98]

[Ali12] Vincent Alimi : *Contribution au déploiement des services mobiles et à l'analyse de la sécurité des transactions*. Thèse de doctorat, Université de Caen Basse-Normandie, 2012. [cité p. 83]

[AMEI98] Dean W Abbott, I Philip Matkovsky et John F Elder IV : An evaluation of high-end data mining tools for fraud detection. *In Systems, Man, and*

Cybernetics, 1998. 1998 IEEE International Conference on, volume 3, pages 2836–2841. IEEE, 1998. [cité p. 129]

[AS02] M. Aglietta et L. Scialom : Les défis de la monnaie électronique pour les banques centrales. *Economies et Sociétés, Série Monnaie, ME*, 4(2):2002, 2002. [cité p. 11, 12, 23]

[AVI03a] AVISPA project : Deliverable 6.1 - list of selected problems. Rapport technique, Avispa, 2003. [cité p. 75]

[AVI03b] AVISPA project : Deliverable d2.1 : The high level protocol specification language. Rapport technique, AVISPA project, 2003. [cité p. 74, 136]

[AVI03c] AVISPA project : Deliverable d6.2 : Specification of the problems in the high-level specification language. Rapport technique, Avispa project, 2003. [cité p. 75, 76, 79]

[AVI06] AVISPA project : Hlpsl tutorial a beginner's guide to modelling and analysing internet security protocols. Rapport technique, AVISPA, 2006. [cité p. 75]

[AW10] Hervé Abdi et Lynne J Williams : Principal component analysis. *Wiley Interdisciplinary Reviews : Computational Statistics*, 2(4):433–459, 2010. [cité p. 114]

[Ban] Bank of England : Frequently asked questions. http://www.bankofengland.co.uk/banknotes/about/faqs.htm. last visited on 20/04/2011. [cité p. 10]

[Ban10] European Central Bank : *The payment system : payments, securities, derivatives and the role of the eurosystem.* European Central Bank, 2010. [cité p. 12, 23]

[Bas98] Basle Committee on Banking Supervision : Risk management for electronic banking and electronic money activities. Rapport technique, Bank for International Settelments, 1998. [cité p. 12]

[BGZ+08] Jason R Beck, Maria Garcia, Mingyu Zhong, Michael Georgiopoulos et Georgios C Anagnostopoulos : A backward adjusting strategy and optimization of the c4. 5 parameters to improve c4. 5's performance. *In FLAIRS Conference*, pages 35–40, 2008. [cité p. 121]

[BH01] Richard J. Bolton et David J. Hand : Unsupervised profiling methods for fraud detection. *In Conference on credit scoring and credit control*, 2001. [cité p. 92, 93, 109]

[BH07] Terri Bradford et Fumiko Hayashi : Complex landscapes : Mobile payments in japan, south korea, and the united states. http://www.kc.frb.org/Publicat/PSR/Briefings/PSR-BriefingSept07.pdf, September 2007. Last visited on 12/04/2013. [cité p. 37]

[BHA+08] Simon Bernard, Laurent Heutte, Sébastien Adam *et al.* : Etude de l'influence des paramètres sur les performances des forêts aléatoires. *In Dixième Colloque International Francophone sur l'Ecrit et le Document*, pages 207–208, 2008. [cité p. 120]

[BJTW11] Siddhartha Bhattacharyya, Sanjeev Jha, Kurian Tharakunnel et J. Christopher Westland : Data mining for credit card fraud : A comparative study. *Decision Support Systems*, 50:602–613, 2011. [cité p. 30, 48, 94, 98]

[BKJ03] E.L. Barse, H. Kvarnstrom et E. Jonsson : Synthesizing test data for fraud detection systems. *In Computer Security Applications Conference, 2003. Proceedings. 19th Annual*, pages 384 – 394. dec. 2003. [cité p. 94]

[BMV05] David Basin, Sebastian Mödersheim et Luca Vigano : Ofmc : A symbolic model checker for security protocols. *International Journal of Information Security*, 4(3):181–208, 2005. [cité p. 74]

[Bou04] Remco R Bouckaert : *Bayesian network classifiers in weka*. Department of Computer Science, University of Waikato, 2004. [cité p. 43, 116]

[Bre01] Leo Breiman : Random forests. *Machine learning*, 45(1):5–32, 2001. [cité p. 45, 115, 116]

[BS03] David Bounie et Sébastien Soriano : La monnaie électronique : Principes, fonctionnement et organisation. *Les cahiers du numérique et des télécommunications*, 4:71–92, 2003. [cité p. 10, 11]

[CBK09] Varun Chandola, Arindam Banerjee et Vipin Kumar : Anomaly detection : A survey. *ACM Comput. Surv.*, 41(3):15 :1–15 :58, juillet 2009. [cité p. 97]

[CCK12] CCK : Quarterly sector statistics report. Rapport technique, Communications Commission of Kenya, 2012. [cité p. 1]

[CG96] Committee on Payment and Settlement Systems et Group Computer Experts of the central banks of the Group of Ten countries : Security of electronic money. Rapport technique, Committee on Payment, 1996. [cité p. 11]

[Cha05] Nitesh V Chawla : Data mining for imbalanced datasets : An overview. *In Data mining and knowledge discovery handbook*, pages 853–867. Springer, 2005. [cité p. 110]

[CHM+10] W. Chen, G. P. Hancke, K. E. Mayes, Y. Lien et J.-H. Chiu : Nfc mobile transactions and authentication based on gsm network. *In Proceedings of the 2010 Second International Workshop on Near Field Communication*, NFC '10, pages 83–89, Washington, DC, USA, 2010. IEEE Computer Society. [cité p. 36]

[CKST01] Suresh Chari, Parviz Kermani, Sean Smith et Ros Tassiulas : Security issues in m-commerce : A usage-based taxonomy. e-commerce agents. *In LNAI*, pages 264–282. Springer, 2001. [cité p. 34, 35, 157]

[CLN09] CasJ.F. Cremers, Pascal Lafourcade et Philippe Nadeau : Comparing state spaces in automatic security protocol analysis. *In Formal to Practical Security*, volume 5458 de *Lecture Notes in Computer Science*, pages 70–94. Springer Berlin Heidelberg, 2009. [cité p. 74]

[CLP06] Laurent Condamin, Jean-Paul Louisot et Naïm Patrick : *Risk quantification - Management, Diagnosis and Hedging*. Wiley, 2006. [cité p. 24, 25]

[Coh60] Jacob Cohen : A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960. [cité p. 110, 111]

[Col10] Hélène Collavizza : *Contribution à la vérification formelle et programmation par contraintes*. Thèse de doctorat, Université de Nice Sophia-Antipolis, 2010. [cité p. 74]

[Com03] Committee on Payment and Settlement Systems : A glossary of terms used in payment and settlement systems. Rapport technique, Bank for International Settlements, 2003. [cité p. 7]

[CP02] Jeffrey Carmichael et Michael Pomerleano : *The development and regulation of non-bank financial institutions*. World Bank Publications, 2002. [cité p. 14]

[CPV01] Corinna Cortes, Daryl Pregibon et Chris Volinsky : Communities of interest. *In In Proceedings of the Fourth International Conference on Advances in Intelligent Data Analysis (IDA*, pages 105–114. 2001. [cité p. 98]

[CT95] John G Cleary et Leonard E Trigg : K^* : An instance-based learner using an entropic distance measure. *In ICML*, pages 108–114, 1995. [cité p. 47, 116]

[DAP09] Linde Delamaire, Hussein Abdou et John Pointon : Credti card fraud and detection techniques : a review. *Banks and Bank systems*, 4:57–68, 2009. [cité p. 30, 48]

[DGB06] Yvo Desmedt, Claude Goutier et Samy Bengio : Special uses and abuses of the fiat-shamir passport protocol (extended abstract). *In* Carl Pomerance, éditeur : *Advances in Cryptology-CRYPTO 87*, volume 293 de *Lecture Notes in Computer Science*, pages 21–39. Springer Berlin Heidelberg, 2006. [cité p. 83, 88]

[DGKN97] Claude Dragon, Didier Geiben, Daniel Kaplan et Gilbert Nailard : *Les moyens de paiement - Des espèces à la monnaie électronique*. GM Consultants associés, 1997. [cité p. 10, 12, 14]

[DMOZ08] Tomi Dahlberg, Niina Mallat, Jan Ondrus et Agnieszka Zmijewska : Past, present and future of mobile payments research : A literature review. *Electronic Commerce Research and Applications*, 7:165–181, 2008. [cité p. 15]

[Doc] Ntt Docomo : Osaifu keitai. http://www.nttdocomo.co.jp/english/service/convenience/osaifu/. Last visited on 12/04/2013. [cité p. 37]

[Eur98] European Central Bank : Report on electronic money. Rapport technique, European Central Bank, 1998. [cité p. 12, 13, 15]

[Eur06] European Union : Directive 2006/48/ce concernant l'accès à l'activité des établissements de crédit et son exercice (refonte), 2006. [cité p. 14]

[Eve92] D Everett : Smart card tutorial. http://www.smartcard.co.uk/tutorials/sct-itsc.pdf, 1992. [cité p. 85]

[FBLR08] Tan Soo Fun, Leau Yu Beng, J. Likoh et R. Roslan : A lightweight and private mobile payment protocol by using mobile network operator. *In Computer and Communication Engineering, 2008. ICCCE 2008. International Conference on*, pages 162 –166, may 2008. [cité p. 35]

[Fed10] Federal Bureau of investigation : Cyber banking fraud. http://www.fbi.gov/news/stories/2010/october/cyber-banking-fraud, January 2010. Last visited on 18/06/2013. [cité p. 100]

[Fer95] J. Ferber : *Multi-agent systems : an introduction to distributed artificial intelligence*, volume 248. Addison-Wesley, 1995. [cité p. 97]

[FP97] T. Fawcett et F. Provost : Adaptive fraud detection. *Data mining and knowledge discovery*, 1(3):291–316, 1997. [cité p. 30, 109]

[FPSS96] U. Fayyad, G. Piatetsky-Shapiro et P. Smyth : From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37, 1996. [cité p. 38, 39, 157]

[Fra02] République Française : Règlement 2002-13 du 21 novembre 2002 relatif à la monnaie électronique et aux établissements de monnaie étlectronique, Novembre 2002. [cité p. 15]

[FW98] Eibe Frank et Ian H Witten : Generating accurate rule sets without global optimization, 1998. [cité p. 46, 116]

[gel] Geldkarte. http://www.geldkarte.de. [cité p. 37]

[GHJV93] Erich Gamma, Richard Helm, Ralph Johnson et John Vlissides : Design patterns : Abstraction and reuse of object-oriented design. *In* Oscar Nierstrasz, éditeur : *Object-Oriented Programming*, volume 707 de *Lecture Notes in Computer Science*, pages 406–431. Springer Berlin, 1993. [cité p. 100]

[GKR+09] J. Gao, V. Kulkarni, H. Ranavat, L. Chang et H. Mei : A 2d barcode-based mobile payment system. *In Multimedia and Ubiquitous Engineering, 2009. MUE'09. Third International Conference on*, pages 320–329. Ieee, 2009. [cité p. 35]

[Glo] Global Platform : Global platform made simple guide : Trusted execution environment (tee) guide. Dernière visite 12/04/2013. [cité p. 33]

[Glo11a] Global Platform : Global platform device secure element remote application management. Rapport technique, Global Platform, 2011. [cité p. 72]

[Glo11b] Global Platform : Tee system architecture, December 2011. [cité p. 33]

[GP10] Chrystel Gaber et Jean-Claude Pailles : Security and trust for mobile phones based on virtualization. *In Norsk Information security conference (NISK)*, 2010. [cité p. 32]

[GWdL08] Manoel Fernando Alonso Gadi, Xidi Wang et Alair Pereira do Lago : Comparison with parametric optimization in credit card fraud detection. *In Machine Learning and Applications, 2008. ICMLA'08. Seventh International Conference on*, pages 279–285. IEEE, 2008. [cité p. 129]

[HAGTR09] JorgeL. Hernandez-Ardieta, AnaI. Gonzalez-Tablas et Benjamin Ramos : Formal validation of ofepsp+ with avispa. *In* Pierpaolo Degano et Luca Vigano, éditeurs : *Foundations and Applications of Security Analysis*, volume 5511 de *Lecture Notes in Computer Science*, pages 124–137. Springer Berlin Heidelberg, 2009. [cité p. 74]

[HB12a] Abdelkrim Hadjij et Madjid Bouabdallah : Secure architecture for off-line mobile payment system. Rapport technique, ORANGE LABS, 2012. [cité p. 66]

[HB12b] Abdelkrim Hadjij et Madjid Bouabdallah : Security requirements of mobile payment systems. Rapport technique, ORANGE LABS, 2012. [cité p. 66]

[HBA+08] Laurent Heutte, Simon Bernard, Sébastien Adam, Émilie Oliveira *et al.* : De la sélection d'arbres de décision dans les forêts aléatoires. *In*

Dixième Colloque International Francophone sur l'Ecrit et le Document, pages 163–168, 2008. [cité p. 45]

[HDO+98] Marti A. Hearst, ST Dumais, E Osman, John Platt et Bernhard Scholkopf : Support vector machines. *Intelligent Systems and their Applications, IEEE*, 13(4):18–28, 1998. [cité p. 44, 45, 116, 157]

[HFH+09] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann et Ian H Witten : The weka data mining software : an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009. [cité p. 114]

[HHH06] Marko Hassinen, Konstantin Hyppoenen et Keijo Haataja : An open, pki-based mobile payment system. *In Emerging Trends in Information and Communication Security*. Springer Berlin / Heidelberg, 2006. [cité p. 35, 36]

[HM08] Constantinos S Hilas et Paris As Mastorocostas : An application of supervised and unsupervised learning approaches to telecommunications fraud detection. *Knowledge-Based Systems*, 21(7):721–726, 2008. [cité p. 109]

[IC07] Jesus Tellez Isaac et Jose Sierra Camara : A secure payment protocol for restricted connectivity scenarios in m-commerce. *In Proceedings of the 8th international conference on E-commerce and web technologies*, EC-Web'07, pages 1–10, Berlin, Heidelberg, 2007. Springer-Verlag. [cité p. 36]

[IETa] IETF Internet Engineering Task Force : Rfc 5996 internet key exchange protocol version 2 (ikev2). [cité p. 52, 54]

[IETb] IETF Network Working Group : Rfc 2865 remote authentication dial in user service (radius). [cité p. 64, 79]

[IETc] IETF Network Working Group : Rfc 3748 extensible authentication protocol (eap). [cité p. 64]

[IETd] IETF Network Working Group : Rfc 5216 extensible authentication protocol tranport layer security (eap-tls) authentication protocol. [cité p. 63]

[IETe] IETF Network Workting Group : Rfc 4301 security architecture for the internet protocol. [cité p. 54]

[IETf] IETF Network Workting Group : Rfc 5281 extensible authentication protocol tunneled transport layer security authenticated protocol version 0. [cité p. 64, 66]

[ISO] ISO : Iso/iec 7816-4 :2013 identification cards – integrated circuit cards – part 4 : Organization, security and commands for interchange. [cité p. 55, 64]

[JL95] George H John et Pat Langley : Estimating continuous distributions in bayesian classifiers. *In Proceedings of the Eleventh conference on Uncertainty*

in artificial intelligence, pages 338–345. Morgan Kaufmann Publishers Inc., 1995. [cité p. 42, 116]

[JSL+05] D.R. Jeske, B. Samadi, P.J. Lin, L. Ye, S. Cox, R. Xiao, T. Younglove, M. Ly, D. Holt et R. Rich : Generation of synthetic data sets for evaluating the accuracy of knowledge discovery systems. *In Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 756–762. ACM, 2005. [cité p. 94]

[JTT10] William Jack, Suri Tavneet et Robert Townsend : Monetary theory and electronic money : Reflections on the kenyan experience. *Economic Quarterly*, 96-1(96):83–122, First Quarter 2010 2010. [cité p. 18, 98, 107, 157]

[Kar04] Stamatis Karnouskos : Mobile payment : A journey through existing procedures and standardization initiatives. *Communications Surveys Tutorials, IEEE*, 6(4):44 –66, 2004. [cité p. 9, 15, 16, 17]

[Key36] Keynes : *Théorie générale de l'emploi, de l'intérêt et de la monnaie*. 1936. [cité p. 10]

[KHVC04] S. Karnouskos, A. Hondroudaki, A. Vilmos et B. Csik : Security, trust and privacy in the secure mobile payment service. *In 3rd International Conference on Mobile Business*, pages 3–5, 2004. [cité p. 35, 36]

[K.J10] K.Julisch : Risk-based payment fraud detection. Research Report, September 2010. [cité p. 30]

[KLK09] Kiran S. Kadambi, Jun Li et Alan H. Karp : Near-field communication-based secure mobile payment service. *In Proceedings of the 11th International Conference on Electronic Commerce*, ICEC '09, pages 142–151, New York, NY, USA, 2009. ACM. [cité p. 36]

[Koh90] T. Kohonen : The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, 1990. [cité p. 43]

[Koh95] Ron Kohavi : The power of decision tables. *In Machine Learning : ECML-95*, pages 174–189. Springer, 1995. [cité p. 46, 116]

[Kok97] A.I. Kokkinaki : On atypical database transactions : identification of probable frauds using machine learning for user profiling. *In Knowledge and Data Engineering Exchange Workshop, 1997. Proceedings*, pages 107–113. IEEE, 1997. [cité p. 97]

[Lan99] Serge Lanskoy : La nature juridique de la monnaie électronique. Rapport technique, Banque Centrale de France, 1999. [cité p. 10, 11]

[LCVH92] Saskia Le Cessie et JC Van Houwelingen : Ridge estimators in logistic regression. *Applied statistics*, pages 191–201, 1992. [cité p. 42, 116]

[Leo95] Kevin J Leonard : The development of a rule based expert system model for fraud alert in consumer credit. *European journal of operational research*, 80(2):350–356, 1995. [cité p. 40]

[LHF05] Niels Landwehr, Mark Hall et Eibe Frank : Logistic model trees. *Machine Learning*, 59(1-2):161–205, 2005. [cité p. 42, 115, 116]

[LIS06] Timothy R. Lyman, Gautam Ivatury et Stefan Staschen : Use of agents in branchless banking for the poor : rewards, risks, and regulation. Focus note 38, CGAP (Consultative Group to Assist the Poor), October 2006. [cité p. 17]

[LK77] J Richard Landis et Gary G Koch : The measurement of observer agreement for categorical data. *biometrics*, pages 159–174, 1977. [cité p. 111, 112, 159]

[LKJ02] Emilie Lundin, Hakan Kvarnström et Erland Jonsson : A synthetic fraud data generation methodology. *In* Robert Deng, Feng Bao, Jianying Zhou et Sihan Qing, éditeurs : *Information and Communications Security*, volume 2513 de *Lecture Notes in Computer Science*, pages 265–277. Springer Berlin Heidelberg, 2002. [cité p. 93, 94, 95, 158]

[Lou05] Jean-Paul Louisot : *Gestion des risques*. AFNOR, 2005. [cité p. 24, 25, 157]

[LPP08] Timothy Lyman, Mark Pickens et David Porteous : Regulating transformational branchless banking. Rapport technique, cgap, 2008. [cité p. 17]

[LRA12] E.A. Lopez-Rojas et S. Axelsson : Multi agent based simulation (mabs) of financial transactions for anti money laundering (aml). *In Nordic Conference on Secure IT Systems*. Blekinge Institute of Technology, 2012. [cité p. 94]

[LTV10] Pascal Lafourcade, Vanessa Terrade et Sylvain Vigier : Comparison of cryptographic verification tools dealing with algebraic properties. *In* Pierpaolo Degano et JoshuaD. Guttman, éditeurs : *Formal Aspects in Security and Trust*, volume 5983 de *Lecture Notes in Computer Science*, pages 173–185. Springer Berlin Heidelberg, 2010. [cité p. 74]

[M8610] M86 Security : Cybercriminals target online banking customers - use trojan and exploit kits to steal funds from major uk financial institution. Rapport technique, M86 Security, 2010. [cité p. 100, 106]

[Mas] Mastercard : Security and risk backgrounder. http://www.mastercard.com/ca/company/en/security_risk.html. last visited on 23/03/2013. [cité p. 48]

[Masre] Massif : D2.3.3 test and evaluation. Rapport technique, MASSIF FP7-257475, 2013 - à paraître en Septembre. [cité p. 99, 136]

[Mat] MATLAB : Chi-square goodness-of-fit test. http://www.mathworks.fr/fr/help/stats/chi2gof.html. Last visited on 18/06/2013. [cité p. 102]

[Mat75] B.W. MATTHEWS : Comparison of the predicted and observed secondary structure of {T4} phage lysozyme. *Biochimica et Biophysica Acta (BBA) - Protein Structure*, 405(2):442 – 451, 1975. [cité p. 110]

[Mes01] Jean-Stéphane MESONNIER : Monnaie électronique et politique monétaire. Rapport technique, Banque de France, 2001. [cité p. 15]

[MM05] Konstantinos MARKANTONAKIS et Keith MAYES : A secure channel protocol for multi-application smart cards based on public key cryptography. *In Communications and Multimedia Security*, pages 79–95. Springer, 2005. [cité p. 85]

[mon] Moneo. http://www.moneo.com/. [cité p. 9, 37]

[Mud13] Joseck Luminzu MUDIRI : Fraud in mobile financial services. Rapport technique, MicroSave, 2013. [cité p. 30]

[MY08] J. MENG et L. YE : Secure mobile payment model based on wap. *In Wireless Communications, Networking and Mobile Computing, 2008. WiCOM'08. 4th International Conference on*, pages 1–4. IEEE, 2008. [cité p. 35]

[Neu] NEURAL TECHNOLOGIES : Minotaurtm fraud detection software - finance sector. http://www.neuralt.com/fraud_detection_software.html. Last visited on 23/03/2013. [cité p. 49]

[NHCV07] M.J. NORTH, T.R. HOWE, N.T. COLLIER et J.R. VOS : A declarative model assembly infrastructure for verification and validation. *In* Shingo TAKAHASHI, David SALLACH et Juliette ROUCHIER, éditeurs : *Advancing Social Simulation : The First World Congress*, pages 129–140. Springer Japan, 2007. [cité p. 100]

[Nou06] Danièle NOUY : Le champs du risque opérationnel dans bâle ii et au-delà. *Revue d'économie financière*, 84(3):11–24, 2006. [cité p. 2]

[NP09] Karsten NOHL et Chris PAGET : Gsm - srsly ? Presented at 26C3 in Berlin, http://events.ccc.de/congress/2009/Fahrplan/attachments/1519_26C3.Karsten.Nohl.GSM.pdf, December 2009. [cité p. 21, 49]

[NSCov] T. N T NGUYEN, P. SHUM et E. H. CHUA : Secure end-to-end mobile payment system. *In Mobile Technology, Applications and Systems, 2005 2nd International Conference on*, pages 4 pp.–4, Nov. [cité p. 36]

[Oku12] Elly OKUTYI : Safaricom tightens security on m-pesa with fraud management system. http://www.humanipo.com/news/1341/Safaricom-tightens-

security-on-M-Pesa-with-Fraud-Management-system, August 2012. Last visit 22/03/2013. [cité p. 48]

[Ora12] Orange : Orange money dépasse les 4 millions de clients et lance ses services en jordanie et à l'ile maurice. http://www.orange.com/fr/presse/communiques/communiques-2012/Orange-Money-depasse-les-4-millions-de-clients-et-lance-ses-services-en-Jordanie-et-a-l-Ile-Maurice, juin 2012. Last visited on 12/04/2013. [cité p. 1]

[PBP+10] Reema Patel, Bhavesh Borisaniya, Avi Patel, Dhiren Patel, Muttukrishnan Rajarajan et Andrea Zisman : Comparative analysis of formal model checking tools for security protocol verification. *In* Natarajan Meghanathan, Selma Boumerdassi, Nabendu Chaki et Dhinaharan Nagamalai, éditeurs : *Recent Trends in Network Security and Applications*, volume 89 de *Communications in Computer and Information Science*, pages 152–163. Springer Berlin Heidelberg, 2010. [cité p. 74]

[Per94] Michel Perdrix : La problématiques des paiements par cartes prépayées. Rapport technique, Banque de France, 1994. [cité p. 11, 12]

[PLSMG05] Clifton Phua, Vincent Lee, Kate Smith-Miles et Ross Gayler : A comprehensive survey of data mining-based fraud detection research. *Artificial Intelligence Review*, 2005. [cité p. 40, 47, 92, 93]

[Por] David Porteous : The global landscape for transformational mobile money. Banca de las Oportunidades Seminar, Bogota, Colombia. Last visited on 11/04/2013. [cité p. 17]

[PWKS11] Yi Peng, Guoxun Wang, Gang Kou et Yong Shi : An empirical study of classification algorithm evaluation for financial risk prediction. *Applied Soft Computing*, 11:2906–2915, march 2011. [cité p. 109]

[Qui93] John Ross Quinlan : *C4. 5 : programs for machine learning*, volume 1. Morgan kaufmann, 1993. [cité p. 45, 116]

[Répa] République Française : Article 1234 code civil. [cité p. 6]

[Répb] République française : Loi bancaire du 24 janvier 1984. [cité p. 6, 14]

[Sim99] Pierre Simon : La position de la profession sur la monnaie électronique. *Revue d'Economie financière*, 53:37–39, 1999. [cité p. 11]

[SKSM08] A. Srivastava, A. Kundu, S. Sural et A.K. Majumdar : Credit card fraud detection using hidden markov model. *Dependable and Secure Computing, IEEE Transactions on*, 5(1):37–48, 2008. [cité p. 43]

[SS12] V.C. Sekhar et M. Sarvabhatla : Secure lightweight mobile payment protocol using symmetric key techniques. *In Computer Communication and Informatics (ICCCI), 2012 International Conference on*, pages 1 –6, jan. 2012. [cité p. 35]

[TCG11] TCG : Tpm main specification. Rapport technique, Trusted Computing Group, 2011. [cité p. 33]

[Tur03] Mathieu Turuani : *Sécurité des protocoles cryptographiques : décidabilité et complexité*. Thèse de doctorat, Henri Poincaré - Nancy 1, 2003. [cité p. 75]

[Tur06] Mathieu Turuani : The cl-atse protocol analyser. *In* Frank Pfenning, éditeur : *Term Rewriting and Applications*, volume 4098 de *Lecture Notes in Computer Science*, pages 277–286. Springer Berlin Heidelberg, 2006. [cité p. 74]

[Tyg96] J. D. Tygar : Atomicity in electronic commerce. *In Proceedings of the fifteenth annual ACM symposium on Principles of distributed computing*, PODC '96, pages 8–26, New York, NY, USA, 1996. ACM. [cité p. 67]

[Uni00] European Union : Directive 2000/46/ce concernant l'accès à l'activité des établissements de monnaie électronique et son exercice ainsi que la surveillance prudentielle de ces établissements, Septembre 2000. [cité p. 12]

[Uni07] European Union : Directive 2007/64/ce concernant les services de paiement dans le marché intérieur, novembre 2007. [cité p. 15]

[Uni09] European Union : Directive 2009/110/ce, 2009. [cité p. 12, 15]

[UPK11] Pascal Urien, Marc Pasquet et Christophe Kiennert : A breakthrough for prepaid payment : End to end token exchange and management using secure ssl channels created by eap-tls smart cards. *In Collaboration Technologies and Systems (CTS), 2011 International Conference on*, pages 476–483. IEEE, 2011. [cité p. 87, 89]

[VDWKP09] Gauthier Van Damme, Karel M. Wouters, Hakan Karahan et Bart Preneel : Offline nfc payments with electronic vouchers. *In Proceedings of the 1st ACM workshop on Networking, systems, and applications for mobile handhelds*, MobiHeld '09, pages 25–30, New York, NY, USA, 2009. ACM. [cité p. 37]

[Vig06] Luca Viganò : Automated security protocol analysis with the avispa tool. *Electronic Notes in Theoretical Computer Science*, 155:61–86, 2006. [cité p. 74]

[VIS] VISA : Security and trust at every level. http://www.visaeurope.com/en/about_us/security.aspx. Last visit on 22/03/2013. [cité p. 48]

[WF05] Ian H Witten et Eibe Frank : *Data Mining : Practical machine learning tools and techniques*. Morgan Kaufmann, 2005. [cité p. 42]

[WHJ+09] Christopher Whitrow, David J Hand, Piotr Juszczak, D Weston et Niall M Adams : Transaction aggregation as a strategy for credit card fraud detection. *Data Mining and Knowledge Discovery*, 18(1):30–55, 2009. [cité p. 114]

[WL90] Bernard Widrow et Michael A Lehr : 30 years of adaptive neural networks : perceptron, madaline, and backpropagation. *Proceedings of the IEEE*, 78(9): 1415–1442, 1990. [cité p. 44, 115, 116, 157]

[WLL+13] Tielei Wang, Kangjie Lu, Long Lu, Simon Chung et Wenke Lee : Jekyll on ios : When benign apps become evil. *In Proceedings of the 22nd USENIX Security Symposium*, 2013. [cité p. 32]

[Wér07] Etienne Wéry : *Paiements et monnaie électronique - Droits européen, français et belge*. Larcier, 2007. [cité p. 6, 8]

[ZFM08] Ge Zhang, Cheng Feng et Christoph Meinel : Simpa : A sip-based mobile payment architecture. *In Proceedings of the Seventh IEEE/ACIS International Conference on Computer and Information Science (icis 2008)*, ICIS '08, pages 287–292, Washington, DC, USA, 2008. IEEE Computer Society. [cité p. 35]

Table des figures

Liste des tableaux

Sécurisation des services de transactions sur terminaux mobiles

---

Les transactions sur mobile suscitent depuis quelques années un intérêt grandissant. Cette thèse se place dans le contexte d'un tel service géré par un opérateur de téléphonie mobile. Les transactions sont réalisées entre souscrivants du service uniquement à l'aide de monnaie électronique privative émise par l'opérateur. Le problème de cette thèse réside dans la sécurisation de ces types de services. Nous proposons dans cette thèse une architecture permettant de garantir une sécurité de bout-en-bout entre l'application et la plateforme de paiement. Celle-ci est basée sur l'utilisation conjoint d'un élément de sécurité *SE* et d'un environnement d'exécution sécurisée *TEE*. Différentes transactions ont été considérées, paiement marchand et transferts entre particuliers, ainsi que différents modes, tout-connecté, déconnecté ou semi-connecté. Les protocoles proposés ont été vérifiés formellement et leurs performances ont été étudiées. Une étude comparative entre différents algorithmes de classification est également réalisée pour les adapter à la détection de la fraude. A cet effet, le système de paiement et le comportement de ses utilisateurs a été modélisé pour créer un générateur de données synthétiques. Une validation préliminaire de ce simulateur a été réalisée. L'originalité du simulateur est qu'il se base sur l'exploitation de données provenant d'un service déployé sur le terrain.

Securing a mobile-based transaction system

Mobile-based transactions have driven growing attention for the past few years. This thesis focuses on mobile-based transaction systems which are managed by a mobile network operator. In such a context, transactions are carried out with electronic money emitted by the operator by the subscribers of the service only. This thesis addresses the problem of securing such services. We propose an architecture which achieves end-to-end security between the payment platform and the payment application in the mobile device. Is is based on a Secure Element (SE) and a Trusted Execution Environment. Several types of transactions were considered such as payments or transfers as well as different modes based on the connection availability of the various actors. The protocols proposed were formally verified. Their performances were also taken into account. Several classification algorithms were compared to be adapted to the fraud detection problem in mobile-based systems. To achieve this, the payment platform and the user's behavior were modeled to create a synthetic data generator. The latter is preliminarily validated in the thesis. The originality of this simulator is that it is based on data from an existing system.

Indexation Rameau (français) : SERVICES MOBILES DE DONNÉES, COMMERCE MOBILE, MONNAIE ÉLECTRONIQUE, SYSTÈMES INFORMATIQUES - MESURES DE SÛRETÉ, CRIMINALITÉ INFORMATIQUE, FRAUDE
Indexation Rameau (anglais) : MOBILE COMMERCE, ELECTRONIC FUNDS TRANSFERS, ELECTRONIC DATA, PROCESSING DEPARTMENTS – SECURITY MEASURES, COMPUTER CRIMES, FRAUD
Indexation libre : paiement sur terminaux mobiles, sécurité de bout-en-bout, classification, détection de fraude, terminaux mobiles, élément sécurisé, environnement d'exécution sécurisée, simulation, modélisation, données synthétiques

---

Spécialité Informatique et applications

Laboratoire GREYC - UMR CNRS 6072 - Université de Caen Basse-Normandie - Ensicaen
6 Boulevard du Maréchal Juin - 14050 CAEN CEDEX