Applying extensions of evidence theory to detect frauds in financial infrastructures

Luigi Coppolino, Salvatore D'Antonio, Valerio Formicola, and Luigi Romano

University of Naples "Parthenope",
Department of Engineering,
Naples, Italy
{luigi.coppolino,salvatore.dantonio,valerio.formicola,lrom}@uniparthneope.it

Abstract

The Dempster-Shafer (DS) theory of evidence has significant weaknesses when dealing with conflicting information sources, as demonstrated by preeminent mathematicians. This problem may invalidate its effectiveness when it is used to implement decision making tools that monitor a great number of parameters and metrics. Indeed, in this case very different estimations are likely to happen, and can produce unfair and biased results. In order to solve these flaws, a number of amendments and extensions of the initial DS model have been proposed in literature. In this work we present a fraud detection system that classifies transactions in a mobile money transfer infrastructure by using the data fusion algorithms derived from these new models. We tested it in a simulated environment that closely mimics a real mobile money transfer infrastructure and its actors. Results show substantial improvements of the performance in terms of true positive and false positive rates with respect to the classical DS theory.

Keywords: Fraud Detection, Mobile Money Transfer, Critical Infrastructure, Dempster-Shafer theory.

1 Introduction

Nowadays mobile communication networks represent a key enabling infrastructure for financial service provision since they offer significant opportunities for increasing the efficiency and pervasiveness of such services by expanding access and lowering transaction costs. Mobile financial services are currently applied to several banking products, such as deposit and transact products, over the counter bill payments, saving products, intra-country remittances, and international remittances. In this paper we focus on Mobile Money Transfer (MMT) services which allow to use virtual money in order to carry out payments, money transfers, and transactions through mobile devices.

Such services are an increasingly important and common payment means in many markets due to the pervasive use of mobile phones, the steady growth in remittances, and the need for an electronic Person-to-Person payment that may be an alternative and reliable option to paper-based mechanisms like cash and checks ([16]).

The growing coverage of cellular networks as well as the increasing availability of mobile communication services are enabling the widespread adoption of mobile-based financial services especially in developing countries, like Kenya, India, Uganda, and the Philippines, thus creating the opportunity for a significant proliferation of mobile commerce services as well as for an expanding financial inclusion. It is expected that in those countries most of mobile financial transactions will concern MMT services in the near term since "unbanked" people, i.e. people who do not have their own bank accounts, will be attracted by financial services allowing for performing payments and remittances by simply using a mobile phone. The same phenomenon is being observed in developed countries where citizens are becoming unbanked due to the widespread economic crisis, and financial service providers are beginning to investigate the potential for adopting these newer payment systems that are emerging in less developed countries in order to meet financial needs of customers.

However, as the mobile financial services market grows, the risks related to the use of mobile phone-based payments increase, since MMT services become the target of more skilled and motivated attackers, and the amazing volume of data impairs the ability of the control mechanisms to timely spot frauds. In one case, a supplier of Mobile Money services lost $3.5 million due to a single type of fraud. Like any other money transfer service, an MMT service is vulnerable to a number of misuses, including money laundering (i.e., disguising the proceeds of crime and illegal activities and transforming them into ostensibly legitimate money or other assets), fraudulent use of customer details, and money theft. More in general, MMT frauds consist in intentional deception performed to gain financial profit.

In this paper, we focus on account takeover in MMT services. There are two main reasons behind this choice. First, account takeover per se is possibly the most prominent fraud in MMT services. Second, it is often the pre-condition for more sophisticated frauds. We propose a Fraud Detection System (FDS) that uses some extensions of the Dempster-Shafer (DS) theory to spot evidence of ongoing account takeover attacks against MMT systems. The DS theory is a data fusion technique that allows to correlate evidence provided by multiple information sources and to compute a belief value. Basically, correlation of attack symptoms through the DS theory-based combination of multiple pieces of evidence significantly outperforms other approaches that use a single source of evidence, thus enabling the proposed FDS to achieve higher detection rates while experiencing a smaller number of false positives. However, in certain cases the DS theory of evidence does not take into account conflicting degree of belief, that means that the experience in conflict is completely discarded, thus generating counterintuitive results. In order to solve this issue, a number of methods and combination rules have been proposed. In this paper we tested and compared these extensions of the DS theory by validating their application to fraud detection in MMT services. In our previous research work [4] we applied the Dempster-Shafer model to MMT services. In that paper we considered an off-line analysis based on the sole Dempster combination rule. In this work we have taken into account other fusion models derived from the initial theoretical framework. In particular the Dezert-Smarandache Theory represents a framework subsuming the initial formulation of DS as a particular case. For what concerns the DS model, the two works - this and [4] - show different performance because we performed an improved tuning in the parameters used for detection. The most significant improvement is obtained with the transaction delay monitor (see Section 3 for more details). In the experimental section we provide a methodology (and a pseudo-code) to assess the performance of those models. Finally, we provide an in-line version of the analyzer that can be used for stream-based processing of events. Due to the lack of real and publicly available MMT service data, the effectiveness of this FDS has been assessed by performing simulations, using synthetic data - containing both legitimate and fraudulent transactions - generated by a simulator which closely mimics the behaviour of a real system, from a major MMT service operator.

The paper is organized as follows. Section 2 provides an overview of the DS theory and presents some extensions of this theory. Section 3 describes the Mobile Money Transfer case study and the architecture of the Fraud Detection System based on the evidence theory. In Section 4 experimental tests and results are presented. Section 5 presents related work on the use of the DS theory and its extensions for fraud and intrusion detection. Finally, Section 6 concludes by remarking achieved results.

2 Extensions of the Demspter-Shafer theory

The main objective of this work is to investigate the performance of different algorithms derived from the Dempster-Shafer's theory of evidence and from subsequent extensions of the initial model. This theory is widely used to perform data fusion, i.e. to obtain a reliable estimation of metrics and parameters representative of the (unknown) state of a system. Informally speaking, by reliable we mean very near to the real value. In our case we want to evaluate its performance on a fraud detection system that uses "features" to classify attacks on transactions made through mobile devices. As we present in this work, the accuracy of a fraud detection system can be significantly improved by considering multiple features, where each of them can represent a different characteristic of the fraudster's behavior. Also, we show that the detection accuracy can be further increased by considering recent modifications to the original mathematical model. These modifications have been elaborated in order to solve problems emerged with the initial mathematical framework which suffered from counterintuitive results under particular conditions. In order to present the advancements of the Dempster-Shafer (DS) theory we recall the basic features of the initial model. The basic principle of the theory of evidence is that it does not require an a priori distribution of the states and knowledge on the system by observers. Indeed, the observer evolves and changes its uncertainty as more observations are realized and more evidence is available. The Dempster-Shafer's Theory (DST) has been introduced in the 1960's by Arthur Dempster ([8]) and then improved with the work of Glenn Shafer ([22]). The theory can be considered as a framework, in that it provides both a theoretical foundation to reason about uncertainty and a set of mathematical tools to work with it. In particular, propositions are subsets of a given set of hypotheses. For example, in a fraud detection system the set of hypotheses is composed of the categories of frauds or not frauds. The events are evaluated based on their propositional set. The sets compose the *frame of discernment* Θ, and, as said, the propositions of interest are in a one-to-one correspondence with the subsets of Θ. In the original DS model the sets of possible states of the system $\theta_{1...}\theta_N \in \Theta$ are mutually exclusive and exhaustive. The exhaustivity property requires that the initial frame of discernment is closed with respect to all the possible propositions, i.e. all the possible sets have been identified. Even if this closure is not done, the model persists by introducing closing elements in the frame of discernment. Another strong assumption of the model is that the sets are exclusive, i.e. their boundaries are clear and not overlapped. This is a strong assumption that - as we will see - can be violated in certain circumstances. In the DS model, Θ has 2^{Θ} subsets. These are called the *power-set* of Θ. The DS model allows to infer the true system state just considering the observations $E_1 \dots E_M$ of the system. Given the evidence E_j, we can assign a probability that this supports a certain hypothesis H_j, i.e. we assign a measure of probability to an element of the power-set. A *basic probability assignment (bpa)* or *basic belief assignment (bba)* is a mass function m which assigns beliefs to a hypothesis or, in other words, the measure of belief that is committed exactly to the hypothesis H. Formally a basic probability assignment is a function $m : 2^{\theta} \to [0, 1]$ such that $m(\emptyset) = 0$ and $m(H) \geq 0, \forall H \subseteq \Theta$ and $\sum_{H \subseteq \Theta} m(H) = 1$. In the DS theory we assign probabilities to elements of the power-set Θ, i.e. to sets. This approach is very different from the the Bayesian one, where we assign probabilities only to single events, i.e. outcomes of the experiments. Again, the bba contains an elementary knowledge (via belief) about the propositions of the frame of discernment. The bba does not provide directly knowledge of individual propositions. This individual knowledge is bounded by the *belief* and the *plausibility* values. These can be calculated as follows:

- the *Belief* function Bel, describing the belief in a hypothesis H, as: $Bel\left(H\right)=\sum_{B\subseteq H}m(B)$. The belief corresponds to the lower bound on the probability or rather measures the minimum uncertainty value about a proposition. Its properties are $Bel\left(\emptyset\right)=0$ and $Bel\left(\Theta\right)=1$.
- the *Plausibility* function of H, Pl(H), which corresponds to the upper bound on the probability and reflects the maximum uncertainty value about proposition H. The plausibility of H is defined as: $Pl\left(H\right)=\sum_{B\cap H\neq 0}m(B)$.

Therefore the true belief in the hypothesis H lies in the interval $\left[Bel\left(H\right),Pl\left(H\right)\right]$, while the degree of uncertainty is represented by the difference $Pl\left(H\right)-Bel\left(H\right)$.

Finally, the DS theory provides a rule of combination that permits to combine two independent pieces of evidence E_1 and E_2 into a single more informative hint:

$$m_{DS}\left(H\right)=\frac{\sum_{B\cap C=H}m_1\left(B\right)m_2\left(C\right)}{\sum_{B\cap C\neq\emptyset}m_1\left(B\right)m_2\left(C\right)} \tag{1}$$

This formula allows to combine our observations to infer the system state based on the values of belief and plausibility functions. The numerator of this equation corresponds to the *conjunctive consensus* - known also as rule of conjunctive combination or Transferable Belief Model (TBM) - on the H set. The denominator of this rule is a normalization factor that takes into account the mass of the agreement among the information sources. It is typically identified as 1-K, where K is the *degree of conflict* among the sources, i.e.:

$$K=\sum_{X_1\cap\ldots\cap X_n=\emptyset}\prod_{i=1}^{n}m_i(X_i) \tag{2}$$

$0\leq K\leq 1$, where 0 means all sources are in full agreement and 1 means all sources are in full disagreement.

Note that the combination rule allows to incorporate new evidence and update our beliefs as new knowledge is acquired. Also, the model allows to combine incomplete, uncertain and also partially contradictory information. The rule does not consider full contradictory sources, because in that case the denominator is zero and no value is defined in the combination rule.

2.1 Counterintuitive results in the Dempster-Shafer model

The denominator in Dempster rule (1-K) normalizes each amount of mass. As a consequence of this normalization, the new combined bba will not take into account the part of conflicting mass, that in other words means that the experience or evaluation in conflict is completely discarded. This produces counterintuitive results in certain cases. This problem was originally pointed out in [26] with an example. Two physicians provide a diagnosis for some neurological symptoms of a patient. The first doctor believes in meningitis with probability of 0.99 or brain tumor with probability of 0.01. The second doctor believes in concussion with probability of 0.99 or a brain tumor with a probability of 0.01. Using the Dempster rule, we find that m(brain tumor) = Bel (brain tumor) = 1. This rule produces a result that both physicians considered to be very unlikely. The generalization of this example can be found in [19]. Another example is provided in [23], where authors talk about *dictatorial power* in the Dempster model, i.e. the model does not conceive the existence of strong sources (e.g. with very small uncertainty) but with different believes.

2.2 Combination rules based on Dempster-Shafer model

In order to solve the problems pointed out in [26], a number of methods and combination procedures have been investigated in literature, mostly addressing the treatment of conflicting evidence and the definition of the frame of discernment. We will discuss some of these alternatives. Some rules are derived from the DS model and some from the extended version of the original theory. Important properties that differentiate the models below can be expressed in the following points: combination results must be coherent for any number of sources, any values of bpa and any types of frames; the rule of combination should preserve the commutativity, i.e. the order the sources are combined should not affect the results; the total ignorance should be neutral with respect to the combination rule, i.e. combining information sources with a new full ignorant source should maintain the same belief; the associativity of the operator. The Dempster rule of combination has all the properties above. Here we recall the Dubois-Prade rule, which is a combination rule preserving associativity and commutativity. Other DS-based rules not discussed here are the Yager's rule, which assigns the conflict to the ignorance, i.e. to the union of all exhaustive and exclusive elements in the frame, and the Smet's rule, which redistributes the conflict to the empty set (i.e. $m_S(\emptyset) \geq 0$), i.e. the empty set is reinterpreted as the set of all not considered hypotheses (not just the null hypothesis). Other methods present in literature perform the averaging of the information sources, where, for instance, each mass is multiplied by a different weight for different sources. A discussion of combination rules based on the Dempster-Shafer model can be found in [20].

2.2.1 Dubois-Prade model and rule

The disjunctive combination rule has been introduced by Dubois & Prade, and has been initially defined on the power-set of the DS model. This rule has been conceived for the case of sources that may be mistaken indifferently. It provides more knowledge when all the sources are conflicting. For two sources it defines $m_{Dj}(\emptyset) = 0$ and for any $X \neq \emptyset$ in 2^{Θ}:

$$m_{Dj}(X) = \sum_{X_1 \cup X_2 = X} m_1(X)\, m_2(X) \tag{3}$$

The rule of combination by Dubois & Prade [11] supposes that in case of conflicts one source is right and one is wrong, while in case of agreement they are both reliable. Thus, in case of agreement, the estimation of the mass is in the intersection of the sets $(X_1 \cap X_2)$, whilst it is in the union in case of conflict $(X_1 \cup X_2)$. The rule is commutative but not associative. The rule defines $m_{DP}(\emptyset) = 0$ and for two information sources and for any $X \neq \emptyset$ in 2^{Θ}:

$$m_{DP}(X) = \sum_{\substack{X_1 \cap X_2 = X \\ X_1 \cap X_2 \neq \emptyset}} m_1(X)\, m_2(X) + \sum_{\substack{X_1 \cup X_2 = X \\ X_1 \cap X_2 = \emptyset}} m_1(X)\, m_2(X) \tag{4}$$

2.3 Dezert-Smarandache's Theory of Plausibility

Dezert and Smarandache point out two main problems in the Dempster-Shafer theory: it is implicitly defined on a finite set of exhaustive and exclusive elements (the power-set), i.e. it is based on the third excluded principle; the limits of the Dempster's rule of combination, as explained above. As for the second problem, several fixing formulas have been proposed in literature, each one with its pros and cons in terms of mathematical properties and applicability. As for the first problem, the principle of the third excluded does not allow hypotheses that can be only vague and imprecise, i.e. it does not take into account situations where precise refinement is impossible to be obtained because exclusive elements cannot be properly identified and precisely separated. But actually this is what happens for a wide class of fusion problems (e.g., in natural language analysis for sets as tallness/smallness, pleasure/pain, cold/hot, Sorites paradoxes, etc.). Many problems of this kind typically identify fuzzy sets. The real nature of the hypothesis reflects on the type of frame of discernment used. In the DS theory the frame is provided by the *power-set*. In order to address other cases two additional sets may be considered: the hyper power-set and the super power-set. The power-set of Dempster-Shafer is composed of the exhaustive elements θ and the elements given by their union, i.e. $2^{\Theta} = (\Theta, \cup)$. The hyper power-set (free Dedekind's lattice) is the base of Dezert-Smarandache theory and is built from union and intersection of hypothesis elements, i.e. $D^{\Theta} = (\Theta, \cup, \cap)$. The super power-set is a power-set when the initial set has to be refined, and is indicated as $S^{\Theta} = (\Theta, \cup, \cap, c(\cdot))$, where $c(\cdot)$ is the complementation. Supposing that the elements in the frame have been refined, the hyper power-set is the most representative of fuzzy sets. Again, note that having refined elements in the frame-set does not mean that the elements are sharply separated, which indeed depends on the nature of the problem. In the Dezert-Smarandache theory (DSmT) it is used to talk about the *free DSm model*. Finally, we report that this relation holds: $\mid 2^{\Theta} \mid = 2^{|\Theta|} \leq \mid D^{\Theta} \mid \leq \mid S^{\Theta} \mid$, i.e. for what concerns the DSmT the number of elements to be considered is larger than for the DST.

2.3.1 PCR5, PCR5-approximate and PCR6

The Proportional Conflict Redistribution (PCR) [9] rule transfers conflicting masses to non-empty sets involved in the conflicts proportionally to the masses assigned to them by sources. The rule works in three steps: calculate the conjunctive rule of the belief masses (see the beginning of this section); calculate the conflicting masses; redistribute the (total or partial) conflicting masses to the non-empty sets involved in the conflicts proportionally with respect to their masses assigned by the sources. The way the conflicting masses are redistributed generated several versions of PCR rules [21]. The most sophisticated one is denoted as PCR5, which is actually the most efficient as claimed by the authors. PCR5 rule is quasi-associative and preserves the neutral impact of the vacuous belief assignment. The rule has been defined for two information sources and states that $m_{PCR5}(\emptyset) = 0$ and for any $X \neq \emptyset$ in D^{Θ}:

$$m_{PCR5}(X) = m_{12}(X) + \sum_{\substack{Y \neq X \in D^{\Theta} \\ X \cap Y = \emptyset}} \left[\frac{m_1(X)^2 m_2(Y)}{m_1(X) + m_2(Y)} + \frac{m_2(X) m_1(Y)^2}{m_2(X) + m_1(Y)} \right] \tag{5}$$

In the formula m_{12} is the consensus and the two terms are used to distribute the conflict mass proportionally. Also, the terms in the internal sum must be discarded if their denominator is zero. The formula can be generalized for more than 2 sources. In this case the PCR mechanism requires the calculation of the whole consensus among the sources and the redistribution of the non-zero conflicts among the sources. This procedure is a combinatory calculation with complexity that grows as more sources are combined. In order to simplify the process, one can reduce the complexity by combining *(s-1)*-th sources with the s-th source. This produces a sub-optimal result because the calculation gives more relevance to the first sources taken into account into the combination; thus the order in which sources are combined affects the result. A more rigorous formulation of PCR5 that allows to extend the PCR rule to any number of sources is the PCR6 proposed by Martin and Osswald [15]. This formula leads to a more algorithmic procedure for its calculation. For s sources is $m_{PCR6}(\emptyset) = 0$ and for any $X \neq \emptyset$ in D^{Θ}:

$$m_{PCR6}(X) = m_{1\ldots s}(X) + \sum_{i=1}^{s} m_i(X)^2 \sum_{\substack{\bigcap_{k=1}^{s-1} Y_{\sigma_i(k)} \cap X \equiv \emptyset \\ (Y_{\sigma_i(1)},\ldots,Y_{\sigma_i(s-1)}) \in (D^\Theta)^{s-1}}} \left(\frac{\prod_{j=1}^{s-1} m_{\sigma_i(j)}(Y_{\sigma_i(j)})}{m_i(X) + \sum_{j=1}^{s-1} m_{\sigma_i(j)}(Y_{\sigma_i(j)})} \right) \tag{6}$$

like the PCR5, the rule holds if $m_i(X) + \sum_{j=1}^{s-1} m_{\sigma_i(j)}(Y_{\sigma_i(j)}) \neq 0$. The function σ_i counts all the elements from 1 to s without i, i.e. $\sigma_i(j) = j$ for $j < i$ and $\sigma_i(j) = j + 1$ for $j \geq i$.

2.3.2 PCR#

In [7] Dambreville proposes a new approach to design the fusion rules based on the *Referee functions*. For the sake of brevity we omit the formulas and the mathematical framework; more details are available in [7] [5]. The Referee function is a function that discriminates the characteristics of all the fusion rules, including those presented by other authors. Its core element is the *conditional arbitrament* quantity used into the generic fusion rule. The *rejection rate* generalizes the conflict mass. The general fusion rule proposed by Dambreville includes a sampling method and a summarization method. The two processes reduce adaptively the set of focal elements, i.e. the set of elements in the frame with non-zero mass. This eventually avoids the combinatorics load by providing an approximation of the most significant bbas. In a first stage, the sources provide the values of beliefs on a specific proposition through a sampling process. Then, the referee function provides a result conditionally to the entries; the final output might not be produced, based on the value of the initial beliefs. The principle is to limit the size of the set of focal elements by reducing this size during the summarization process. According to Dambreville, the PCR6 algorithm works just in case of full consensus or no-consensus, but no intermediate cases are considered. The author proposes a new rule, named PCR#, which is able to address the above mentioned case.

3 Use of the Dempster-Shafer Theory and its extensions for Fraud Detection: The Mobile Money Transfer Case Study

3.1 MMTS infrastructure

A Mobile Money Transfer (MMT) service relies on the use of virtual money, called *mMoney*, to perform various types of money transfers and transactions. For example, a customer can use his/her mobile phone for purchasing goods, receiving his/her salary, paying bills, taking loans, paying taxes or receiving social benefits. MMT services are becoming more and more appealing, especially in developing countries, where banking infrastructures are not as capillary as in developed ones, whereas the penetration of mobile phones is high (as compared to bank accounts), and the regulatory environment is weak. In these countries, the number of customers is increasing at a fast pace. According to the 2012 Global Mobile Money Adoption Survey on the status of the mobile money industry ([18]), 150 live mobile money services for unbanked people were active in 2012, 41 of which were launched in 2012. Almost 30 million

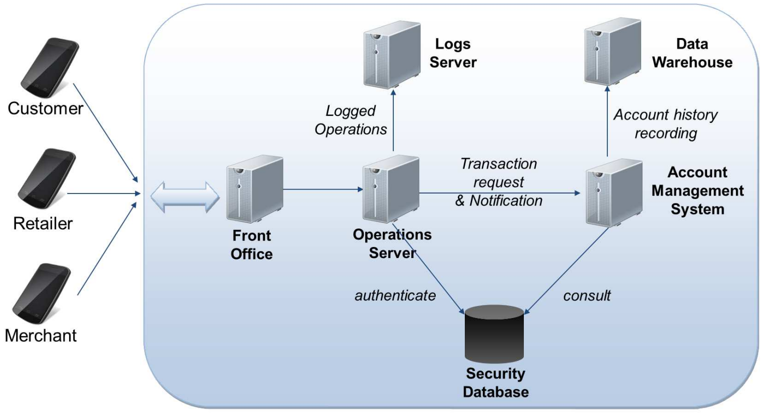

Figure 1: Architecture of a Mobile Money Transfer system.

active customers used mobile money services, who performed 224.2 million transactions totaling $4.6 billion during the month of June 2012. M-Pesa, which was launched in 2007 in Kenya, totaled in December 2011 about 19 million subscribers, i.e. 70% of all mobile subscribers in Kenya ([2]).

Like any other money transfer service, an MMT service is vulnerable to a number of fraud schemes. Fraud is commonly understood as dishonesty calculated for advantage, i.e. deception deliberately practiced in order to secure unfair or unlawful gain. In the context of mobile money fraud is the intentional and deliberate action undertaken by players in the mobile financial services ecosystem aimed at deriving gain (in cash or e-money), and/or denying other players revenue and/or damaging the reputation of the other stakeholders. Furthermore, as telecommunication operators support the provision of financial services across shared networks in cross-border jurisdictions, the large adoption of mobile payment services can result in a growing risk of money laundering in mobile transfer services. Since the success of any payment system is based on ubiquity, convenience, and trust, it is necessary to address emerging security risks in order to safeguard public confidence in MMT services. Fraud detection is particularly challenging in the MMT context due to the scarce willingness to disclose security information by mobile service providers as well as due to the confidentiality requirement that has to be met while dealing with user profiles and transaction data.

As depicted in Figure 1, a MMT infrastructure is normally composed of the following subsystems: the Front Office (FO) that authenticates the users - through the Operations Server - and forwards requests for performing financial transactions to the Account Management System; (AMS); the AMS that authorizes and processes the transactions; the Security Database that contains the security information related to MMT service users (thresholds, blocked accounts, activated/deactivated accounts, number of transactions within a certain time period); the Logs Server that stores logs related to the operations performed by the MMT system; the Data Warehouse that contains historical data about user's activities and accounts. Both the FO and the AMS query the Security Database to authenticate users and manage transactions.

In a MMT service scenario each user of the system has some virtual money that he/she can use to perform various types of money transfers and transactions. Mobile money service users comprise customers, retailers of *mMoney*, and merchants. These actors use their mobile phones to communicate with the FO that provides the interface towards the Operations Server. Each user is an *mWallet holder*. An *mWallet* is an account hosted in the system enabling the *mWallet holder* to carry out various actions by using *mMoney*. In order to access the system MMT service users are required to connect to the FO and authenticate to the Operations Server. This server is in charge of authenticating users, executing simple account management operations (e.g. PIN code update), and delivering notification messages. The Operations Server provides two main functions: view through a User Interface, i.e. the Operations Server interacts with the users to collect operation requests and send notifications; Processing of an operation request, i.e. the Operations Server analyzes the request coming from the user and implements the actions needed to fulfill that request. The server either performs the operation by itself (this is the case when, for example, the requested action consists in modifying customer's password for the service or authenticating the user) or forwards the request to the Account Management System (this happens when the operation concerns account management or credit/debit control). The Account Management System is in charge of managing accounts. In particular it controls user's credit/debit before a financial transaction is authorized and performed. Furthermore it stores information about users' habits.

The Operations Server is also linked to the Logs Server that stores logs of any operation carried out in the system. Logs contain records of users' activities, such as requests for PIN modification, failed authentication, transaction requests, notifications of successful transactions. The input to the MMT system is an operation request received from *mWallet holders*, while the output is the notification of success/failure of the required operation, that implies the registration of operation information. Data that are of interest to fraud detection activities are archived in the Logs Server and in the Data Warehouse. While the security-relevant information that can be gathered by accessing the Logs Server and parsing the stored logs can be used to detect simple fraud cases, historical data about accounts available in the Data Warehouse can be exploited to draw customer behavior and support the detection of complex frauds.

The MMT misuse case addressed in the paper is called Account Takeover. That misuse case is particularly challenging because the attacker uses stolen credentials to perform a violation, thus making it difficult to detect the anomaly at infrastructural level, like, for instance, analyzing network packets or the execution of suspicious applications. This misuse case relies on the following scenario: a fraudster steals the mobile phone of a legitimate MMT service customer and uses it to make illicit money transfers. It is very likely that the behavior of the fraudster differs from that of the legitimate user. In order to detect such fraudulent behavior we exploit data fusion techniques based on the theory of evidence. Specifically, we test several algorithms - introduced in the previous section - to combine the metrics of attack and design a detector of anomalous behavioral patterns. The detector compares the customer behavior with a normal user's profile.

3.2 A Fraud Detection System based on evidence theories

In this section we present a Fraud Detection System (FDS) based on the theories of evidence and plausibility introduced in the previous section. Our objective is to evaluate the performance of this detector by investigating how different algorithms behave under different assumptions by an expert system. We assess the performance of different algorithms with different bbas. Also, we investigate the performance as a function of several detection parameters, specifically the thresholds to discriminate the belief of attacks. In this section we describe the general work flow of the detector and provide details on the monitors of single features, i.e. how the bbas have been assigned by the experts. The detector uses a number of rules that analyse the deviation of each incoming transaction from the normal profile of the user. The rules assign beliefs to ”features”

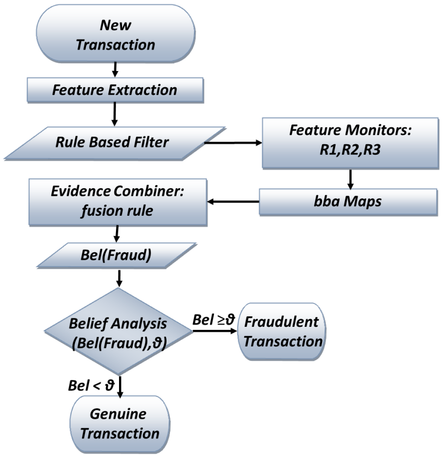

Figure 2: Flow diagram of the proposed FDS.

of the transactions. The belief values are combined to obtain an overall belief by applying the Dempster-Shafer model (DS), the Dubois&Prade models (DP), the Dezert-Smarandache model with several versions of the PCR algorithms (PCR5a, PCR6, PCR#). The overall belief is then compared with a "detection" threshold in order to understand if the user's behaviour has to be considered fraudulent or genuine. The proposed FDS consists of the following three major components: Rule Based Filter, Evidence Combiner, and Analyser. The flow of events in the FDS has been depicted in the flow diagram in Figure 2. To evaluate the effectiveness of the proposed FDS we generated - via accurate simulation of a real system - events representing an account takeover misuse case, where a fraudster steals the mobile phone of a legitimate user and uses it to make money transfers. Particularly the fraudster chooses a victim and approaches his/her person. Once in touch with the victim, he/she steals the phone. Then he/she tries to guess the PIN related to the mobile payment application. Usually the fraudster makes several attempts to access the victim's account. After breaking in the system, he/she typically starts to purchase goods from various merchants.

3.2.1 Rule Based Filter (RBF)

The RBF is a rule-based module that classifies the transactions executed by the users and assigns a certain rank of the fraud risk to them. The assigned value is a measure of the deviation of the observed behavior from the normal behavior profile. The rules used in our study are:

- Rule R1, number of authentication attempts: we analyzed the number of authentication attempts performed by the regular users of the system and by the fraudsters. The larger the number of attempts, the higher is the probability that the transaction is fraudulent.
- Rule R2, delay of authentication attempts: we analyze the time interval between the first and the last authentication attempt being failed. If this time interval exceeds a given threshold (e.g. 15 seconds), then there is a high probability that the transaction is fraudulent.
- Rule R3, outlier detection: users usually carry out similar types of transactions in terms of amount. Supposing we build a cluster of regular transactions, fraudsters are likely to perform transactions out of this cluster. This process is known as outlier detection.

An outlier detector must take into account the transaction amount, the date, and the identification code of the customer. As we describe later, in the experimental trials every user spends money following a Normal distribution $N(\mu, \sigma^2)$ with mean μ and standard deviation σ^2. We can define an average distance of the transaction amount from the average amount spent. To do so, we calculate the area under the Gaussian curve. This value (which is actually a probability) is used to model the possibility that a transaction is a fraud.

Normally an FDS is subjected to a large number of transactions, mostly genuine. The role of the RBF is essential since it separates out most of the easily recognizable genuine transactions from the rest.

3.2.2 Evidence Combiner (EC)

The EC combines the bbas resulting from the application of rules R1, R2, and R3 and computes the overall belief of attack for each transaction. For the detection of frauds in the MMT system, the DS, DP, PCRs allow to introduce alternative sets and a rule for computing the confidence levels associated to them. In order to apply the rules of combination we need to define a frame of discernment U which is a set of exhaustive possibilities. Note that we do not impose the exclusivity in our model, i.e. we consider the general case that the possibilities compose a power-set as used in the DST or a hyper power-set as used in the DSmT. The refined elements in the frame are $U = \{fraud, \neg fraud\}$. In the following F is fraud, N is not-fraud and S is ignorance, i.e. $F \cup N$. In our experimental campaign - presented in the next section - we perform exhaustive research of the best bba combination. In particular we created several bbas for each Rule detector and tested the performance of each algorithm by combining all of them. In the end, our experiments lead to the best performance in terms of bbas and fusion algorithms. In the following we reported the bbas we used for the three rules R1, R2, R3. The notation $m_j(i)$*(N,F,S)* indicates the i-th vector of masses for the rule j, and the three variables N, F, S are the masses committed in that vector to Not Fraud, Fraud and Ignorance:

- mass probability m_1. Let c denote the number of attempts made by a user to access the system, we defined the vector of assignments $m_1(i)$*(N,F,S)* for different values of c, and i is an index in the vector of each bba:

$$bpa(m_1): \begin{cases} \text{c=0} & m_1(0) = (0.1, 0.7, 0.2) \\ & m_1(1) = (0.15, 0.6, 0.25)\ m_1(2) = (0.05, 0.65, 0.3) \\ \text{c=1} & m_1(0) = (0.35, 0.45, 0.2) \\ & m_1(1) = (0.3, 0.45, 0.25)\ m_1(2) = (0.15, 0.50, 0.35) \\ \text{c=2} & m_1(0) = (0.55, 0.3, 0.15) \\ & m_1(1) = (0.6, 0.35, 0.05)\ m_1(2) = (0.25, 0.35, 0.4) \\ \text{c=3} & m_1(0) = (0.7, 0.1, 0.2) \\ & m_1(1) = (0.7, 0.15, 0.15)\ m_1(2) = (0.4, 0.15, 0.45) \\ \text{c>3} & m_1(0) = (0.85, 0.1, 0.05) \\ & m_1(1) = (0.85, 0.05, 0.1)\ m_1(2) = (0.55, 0.05, 0.4) \end{cases}$$

- mass probability m_2. Let t denote the time interval between the first and the last authentication attempt, we adopted the following assignments, $m_2(j)$ for different values of t. Also, we introduced a Δ factor to scale the thresholds. Note that the 0 value of delay is a starting condition for the Rule, since it represents the first transaction/authentication attempt performed by the user; for this reason it is assigned a neutral value with respect to the evidence theory, i.e. full ignorance (0,0,1):

$$bpa(m_2):\begin{cases}\text{t>60}\cdot\Delta \qquad m_2(0)=(0.2,0.6,0.2) \quad m_2(1)=(0.1,0.65,0.35)\\ \qquad m_2(2)=(0.1,0.7,0.2)\\ 5\cdot\Delta<t<60\cdot\Delta \qquad m_2(0)=(0.5,0.3,0.2) \quad m_2(1)=(0.4,0.4,0.2)\\ \qquad m_2(2)=(0.5,0.2,0.3)\\ \text{t<5}\cdot\Delta \qquad m_2(0)=(0.75,0.1,0.15) \quad m_2(1)=(0.8,0.1,0.1)\\ \qquad m_2(2)=(0.8,0.1,0.1)\\ \text{t=0}\cdot\Delta \qquad m_2(0)=(0,0,1) \quad m_2(1)=(0,0,1)\\ \qquad m_2(2)=(0,0,1)\end{cases}$$

- mass probability m_3. This feature deals with the capability of the detector to discriminate regular users from fraudsters through the transaction amounts. The effectiveness of this feature is mostly due to the capability of modeling the spending attitude of fraudsters and regular users correctly. As we will see in the experimental section, MMT operators are usual to model regular users with a stable Gaussian distribution, i.e. the parameters characterizing this distribution are stationary. On the other hand, the fraudster characterization can take into account several distinct profiles. In the case of a mean thief, it is very common that he/she performs small transactions in a short time, as soon as he has stolen the victim's mobile. For this reason we calculate for any transaction the mass probability using the area under the Gaussian curve associated with the expenses of the user. Let a denote the amount of the transaction, the area has been calculated using the Cumulative Distribution Function of the Normal distribution $cdf(a)$ associated to a specific user. We use the amount $\nu = abs(1-2cdf(a))$ of a given Normal distribution to represent the distance of the amount from the average value. Based on the specifics of the MMT operator, for values of probability lower than 2/3, the transaction amount provides uncertain evidence of fraud. More details on the characterization of the users and fraudsters are available in the next section. For high amounts, we give more evidence to the legitimacy of the transactions. Hence, we consider the following assignments to m_3*(N,F,S)*:

$$bpa(m_3):\begin{cases}\nu<0.66 \qquad m_3=(0.6,0.2,0.2)\\ \nu\geq 0.66 \qquad m_3=(0.85,0.05,0.1)\end{cases}$$

As we can see, the values provided by the three features can produce high conflicts, which are not easy to be solved with the classic DST model.

3.2.3 Analyser

In this module we perform the analysis of the fused bbas of the three features. The three bbas are combined using the formulas in the previous section, and provide the value of *Belief* (Bel) of Fraud (F) and Not Fraud (NF) for each authentication attempt as well as for each transaction made by the users. Particularly, we consider *Bel(F)* as the minimum probability that the event "*Fraud*" occurs. In order to classify the events, we define a "detection" threshold θ, where $0 \leq \theta \leq 1$. Single threshold based classification is commonly used in the DS theory. Specifically, if $Bel\,(F) < \theta$ the user behavior is considered genuine and the FDS does not

User type	*Actors*	*PIN*	*Time*	*Amount*
Regular User	200	*N(0,0.35)*	*N(15,10)*	*N(50,30)*
Fraudster	3	*U(0,10)*	*U(1,10)*	*U(31,50)*

Table 1: Simulation parameters. N and U are Normal and Uniform distribution respectively.

generate alarms. Conversely, if $Bel\,(F) \geq \theta$ the event is considered suspicious and the system generates an alarm.

4 Experimental campaign

The objective of this experimental campaign is to evaluate the performance of our FDS with different data fusion rules. Also, we perform the tuning of bpas in order to obtain the best performance for each algorithm. Due to unavailability of real samples related to frauds, we used a simulator that closely mimics the behavior of the real system to create data related to several thousands of transactions of the mobile money transfer service (in the experiments we focused on transactions executed in a delimited geographical area and in a relatively short time). The simulator ([12]) reproduces the operations of the real system in great detail, including virtual money exchange operations by m-vendors, log collection systems, authentication servers and transaction authorization servers, and more. These entities generate a multitude of logs that contain authentication records, money transfer logs, real to virtual currency conversion operations, etc.. The simulator creates events related to legitimate users and to fraudsters. The simulator can be configured in order to define the number of legitimate users, fraudsters, (virtual money) merchants, m-vendors, and to generate random lists of customers' preferred merchants. System activities are driven by random processes. User behavior is given in Table 1. Regular users and fraudsters enter the PIN in the system to perform transactions. Legitimate users rarely make wrong authentication attempts: PIN error distribution is a Normal with 0 mean and 0.35 standard deviation. Fraudsters are more prone to PIN errors that has been modeled as Uniform distribution in the range 0-10. Legitimate users successfully entering the system perform transactions following a *N(50,30)* distribution. Fraudsters perform transactions based on Uniform distribution in the 31-50 range. These values are proved to be very closely to the behavior of mean users and fraudsters. Finally, the time between two distinct authentication attempts or between transaction executions is modeled as a *N(15,10)* distribution - where 15 is expressed in time units (e.g. seconds of simulation run) - for legitimate users. Fraudsters attempt PINs and perform transactions with Uniform distribution between 1 and 10 time units.

Before analyzing the performance of the combination rules, we shortly describe the pseudo-code of the detector and of the performance evaluator. The code is provided in the algorithms shown in blocks 1 and 2. The core of the implementation is the application of the fusion rules discussed in Section 2. We implemented the fusion rules by adapting the library described in [6]. Also, the implementation of PCR5 and PCR5 approximated is new since it is not provided in this template library. The implementations are in Java programming language. It is worth noting that the current implementation of Dubois-Prade model in that library does not consider intermediate cases. During the experiments the DP model exposed the worst performance, and we think this is consequence of the current implementation in use.

In block 1 we provide the procedure used for the performance assessment. This code is similar to that used in our previous work [4], except that there we considered only the DS model for the fraud detection. Also we point out that performance in this work is better than

Algorithm 1 Fraud Detection System Performance Assessment Procedure

```
procedure ASSESS_DETECTION($Event\_Set$)
    Algorithms[]={DS,DP,PCR5a,PCR6,PCR6#}
    for $\Delta = 0.0$ to $2.0$ step $0.2$ do
        for each bba_array $m_1[]$ in $m_1[][]$ do
            for each bba_array $m_2[]$ in $m_2[][]$ do
                for $\theta = 0$ to $1$ step $0.1$ do
                    TP[], TN[], FP[], FN[] = 0

                    for each $New\_Event\_Log$ in $Event\_Set$ do
                        tokens[] = extract_tokens($New\_Event\_Log$)
                        $userID$ = get_userID(tokens[])
                        features[] = extract_features(tokens[])
                        $transaction\_profile$ = get_transaction_profile($userID$)
                        update_users_profile($userID$, $tokens$[])
                        $GroundTruth$ = get_ground_truth($tokens$[])
                        $m_1$ = get_current_m1($m_1[]$, $features[1]$)
                        $m_2$ = get_current_m2($m_2[]$, $\Delta$, $features[2]$)
                        $m_3$ = get_current_m3( $features[3]$, $transaction\_profile$ )
                        for each $Algorithm$ in Algorithms[] do
                            $i$=$Algorithm$
                            $m_{fused}[i]$ = combine($i$, $m_1$, $m_2$, $m_3$)
                            if $m_{fused}[i] < \theta$ then
                                if $GroundTruth$==FRAUD then
                                    FN[$i$]=FN[$i$]++
                                else
                                    TN[$i$]=TN[$i$]++
                                end if
                            else
                                if $GroundTruth$==NOT_FRAUD then
                                    FP[$i$]=FP[$i$]++
                                else
                                    TP[$i$]=TP[$i$]++
                                end if
                            end if
                        end for
                    end for

                    TPR[$i$] = calculateTPR(FP[$i$],FN[$i$],TP[$i$],TN[$i$])
                    TNR[$i$] = calculateTNR(FP[$i$],FN[$i$],TP[$i$],TN[$i$])
                    FPR[$i$] = 1-TNR[$i$]
                    FNR[$i$] = 1-TPR[$i$]
                    StoreForROC($\theta$, $\Delta$, $i$, TPR[$i$], FPR[$i$])

                end for
            end for
        end for
    end for
end procedure
```

Algorithm 2 In-Line Fraud Detection Process

```
 1: procedure PERFORM_DETECTION($New\_Event\_Log$)
 2:     Algorithms[]={DS,DP,PCR5a,PCR6,PCR6#}
 3:     tokens[] = extract_tokens($New\_Event\_Log$)
 4:     $userID$ = get_userID(tokens[])
 5:     features[] = extract_features(tokens[])
 6:     $transaction\_profile$ = get_transaction_profile($userID$)
 7:     update_users_profile($userID$, $tokens$[])
 8:
 9:     $m_1$[] = get_counter_profile($userID$)
10:     $m_2$[] = get_delay_profile($userID$)
11:     $\Delta_{user}$ = get_delta_profile($userID$)
12:     $m_1$ = get_current_m1($m_1$[], $features$[1])
13:     $m_2$ = get_current_m2($m_2$[], $\Delta_{user}$, $features$[2])
14:     $m_3$ = get_current_m3( $features$[3], $transaction\_profile$ )
15:     for each $Algorithm$ in Algorithms[] do
16:         $m_{fused}[Algorithm]$ = combine($Algorithm$, $m_1$, $m_2$, $m_3$)
17:     end for
18:     for each $m_{fused}$ in $m_{fused}$[] do
19:         if $m_{fused} \geq \Delta_{user}$ then
20:             AlarmVector[$Algorithm$] = 1
21:         else
22:             AlarmVector[$Algorithm$] = 0
23:         end if
24:     end for
25:     Push(AlarmVector[])
26: end procedure
```

in [4], because we performed more fine-grain tuning on the parameters used for the detectors. The most significant improvement is obtained from the transaction delay monitor - the Rule 2 - which now has been parametrized on a scale factor Δ. All the tests are performed by considering the *ground truth* of the events, i.e. the events contain information about the actual malicious state of the activity (Fraud or Not Fraud) performed by the regular user/fraudster. Ground truth labeling can be easy done through a simulator, because for each activity we set the label during the logging process based on the real identity of the actor. In both algorithms once a log is entered, it is parsed and tokenized in order to extract the information about the event. Specifically, we are interested in $UserID$, $TransactionAmount$, $Timestamp$. The retrieving process is in lines 10-18 for block 1 and in lines 3-11 for block 2. Note that in block 1 line 15 is used to obtain the ground truth, which is not present in the in-line process, where the ground truth is the hidden state to be discovered. In both algorithms for each user we retrieve the behavior profile related to the three features. This can be provided, for instance, by analytics done on user habits. In the case of a simulated scenario, we suppose the user's behavior is known and is given by the stochastic distribution provided in Table 1. This approximation is correct because we set up the scenario following those parameters.

For what concerns the in-line procedure in block 2, we consider the general case that the threshold θ is different for each user. Again, in a simulated scenario with similar users, we can consider the same θ for any actors. In lines 12-14 we apply the detection rules described in the previous section. Given the specific bba for that user and the current value of the feature, we obtain the mass of the attack state (m_1,m_2,m_3) for the three features. Then we combine the masses using different rules (line 15-17) and store the value in a vector of masses. In the final loop we scan the masses to see if they exceed the alarm threshold θ. The final output of the in-line detector 2 is a Vector of Alarm flags. Additional combinations might be performed on this vector (e.g. majority voting), but are out of the scope of this paper.

The approach to performance assessment in block 1 is similar, but the analysis has to be done on the whole set of events. In block 1 the performance is calculated by scanning linearly all the detection parameters indicated in the previous sections (lines 3-6), namely: 1) the Δ of the delay feature detector; 2) the two bpa vectors - counter and delay - to be combined with the fusion algorithms; 3) the θ threshold. Also, for each combination of parameters, data fusion rules are applied to produce a vector of masses ($m_{fused}[]$) in line 21. Lines 22-34 perform the comparison between the triggered alarm and the ground truth. Finally, rates are calculated in 38-42 and stored in conjunction with the specific parameters that lead to those values.
As shown in the algorithm in block 1, in order to evaluate the FDS, we tested the performance of the classifier with different values of the θ threshold. θ ranges between 0 and 1 and is used to decide whether a given transaction/authentication attempt is to be considered dangerous or not, based on the Bel value of the event. Bel is calculated by combining the bpa of the three features. If the authentication fails, we consider only the Bel calculated starting from Rule R1 and Rule R2, i.e. the amounts have not been considered. Also, for each θ, we calculated four typical classification metrics: True Positives (TP), True Negatives (TN), False Positives (FP), False Negatives (FN). From these metrics, two indexes have been calculated, namely the True Positive Rate (TPR), i.e. TP/(TP+FN), and the False Positive Rate (FPR), i.e. FP/(FP+TN). Both have been plotted on ROC curves. The ROC curves have been plotted for all the combination rules in Figures 3-11. The curves are related to the Δ factor equal to 0.200, which is the one that provided the best values for the detectors. The ROC curves show that the DS combination rule changes smoothly with respect to the others when FPR has lower values. Instead, the best values are reached by other combination rules. In particular, the curves are obtained considering all the combinations of bpas for Rule 1 and Rule 2, namely changing in turns $m_1()$ and $m_2()$ vectors. The ROCs show that $m_2(2)$ significantly contributes to improve the performance of the whole detector for any combination with $m_1()$, and this effect is particularly evident in the DP case. A possible explanation for this result is that $m_2(2)$ reflects less ignorance and more confidence in detecting fraudsters with respect to PIN counter and transaction amount detectors. In short, given our model of fraudster, the transaction delay is the more significative parameter to detect fraudsters. In Table 2 we have reported the three highest values of TPR and FPR reached by each combination rule. In particular, we have given priority to cases where TPR $>$ 99% and FPR $<$ 10%. Also in the second column we have indicated how many times those specific values are reached in the performance dataset provided by the algorithm 1. Note that there are 99 assessments performed per algorithm (for each Δ). As indicated in Table 2, the best TPR/FPR ratio is given by the PCR5 approximated version, while the best top three measurements are provided by the PCR#. The DS is probably the most stable with respect to the θ since it shows small variations on the TPR/FPR ratio for several values. Also note the sharp variation of the DP algorithm, which suddenly falls in performance. This means that if we do not properly choose the bpas for DP, we can have a dramatic loss of performance. Instead, the DP combination gives the highest TPR, which is 99.88 %. We observe from the ROCs that the DP algorithm shows the worst performance for $m_2(0)$. We guess that this happens when the two bpas for Rule 1 and Rule 2 are too much similar, and this is not a good choice for the DP algorithm; indeed, this model has been conceived for diverging

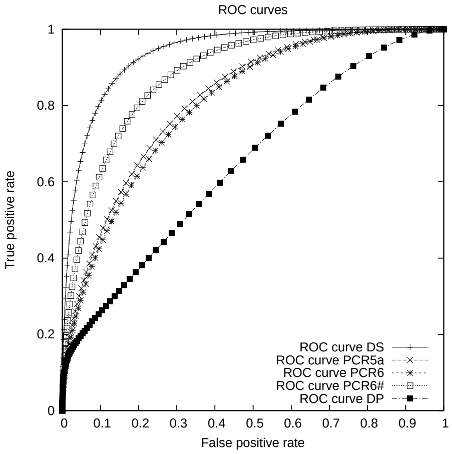

Figure 3: ROC curves for bpas $m_1(0)$ and $m_2(0)$ and m_3. Δ scale factor is 0.200

Algorithm	**N. of times (on 99)**	**TPR %**	**FPR %**	**Tot.**
DS	2, 1, 10	99.28, 99.28, 98.93	6.28, 6.53, 5.53	20
DP	2, 1, 5	99.88, 99.88, 92.49	7.09, 7.09, 7.09	3
PCR5a	7, 2, 3	97.38, 97.38, 89.99	0.52, 0.77, 0.52	12
PCR6	3, 1, 2	98.93, 97.38, 97.38	5.53, 0.52, 0.77	6
PCR6#	3, 8, 3	99.28, 99.28, 98.93	6.28, 6.53, 5.53	21

Table 2: Best Performance per algorithm. Criteria: $TPR > 99\%$, $FPR < 10\%$.
If no $TPR > 99\%$ exists, the highest TPRs have been selected.
The last column is the total times TPRs exceed 99%.

bpas. As said, the $m_2(2)$ produces the best ROCs; this means that the bpa of the delay feature is approximating the maximum (in terms of TPR and FPR) on the $m_2(2)$ value. In Table 3 we have reported the five best TPR and FPR couples and the corresponding indexes for m_1 and m_2 that resulted in those values. We note that this is true in terms of global behavior of the detector. Instead, the maximum value for the combination rules does not happen necessarily on the $m_2(2)$ case. As for m_1 we observe that the best ROC curve is for DS when we consider $m_1(0)$ - and $m_2(2)$. Instead, the other curves ($m_1(1)$, $m_1(2)$) lower the performance of DS. The most interesting aspect here is that the other combination rules stay more or less on the same values as we change m_1. This can be seen like a sort of robustness of the other rules with respect to the DS combination rule, as we change the bpas. In the last column of Table 3 we have indicated how many times all the best values are reached by the algorithms. The algorithm with the most frequent high performance is the PCR# followed by DS. Finally, we want to point out the reason why DS rule is still good in most of the cases. The reason is that our detectors generate conflicting masses, but the occurrence of conflict is a rare event in our current dataset. This means that on average the DS is able to generate good decisions, but for particular and rare cases, it is not the best detector. Instead, the PCR is able to preserve the DS performance and even to improve such performance in those rare situations.

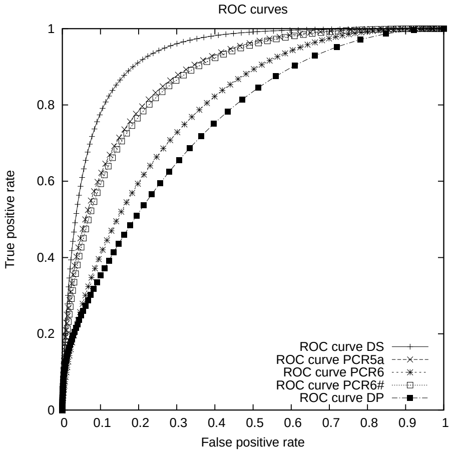

Figure 4: ROC curves for bpas $m_1(0)$ and $m_2(1)$ and m_3. Δ scale factor is 0.200

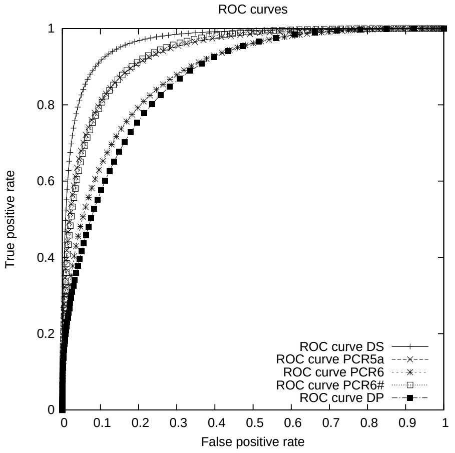

Figure 5: ROC curves for bpas $m_1(0)$ and $m_2(2)$ and m_3. Δ scale factor is 0.200

Algorithm	**MapID**	Δ	θ **threshold**	**TPR %**	**FPR %**
DP	(0,1)	0.200	0.2	99.88	7.09
DP	(0,1)	0.200	0.4	99.28	6.28
DS	(0,0)(0,1)	0.200	0.5-0.4-0.4	99.28	6.28
$PCR6\#$	(0,0)(0,2)	0.200	0.5-0.4	99.28	6.53
DS	(0,2)	0.200	0.4	99.28	6.53

Table 3: Best Performance with $TPR > 99\%$ and $FPR < 10\%$.

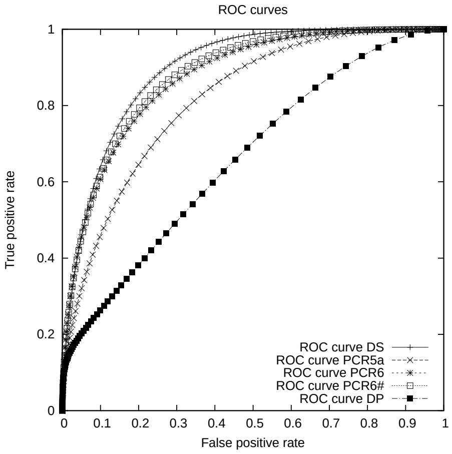

Figure 6: ROC curves for bpas $m_1(1)$ and $m_2(0)$ and m_3. Δ scale factor is 0.200

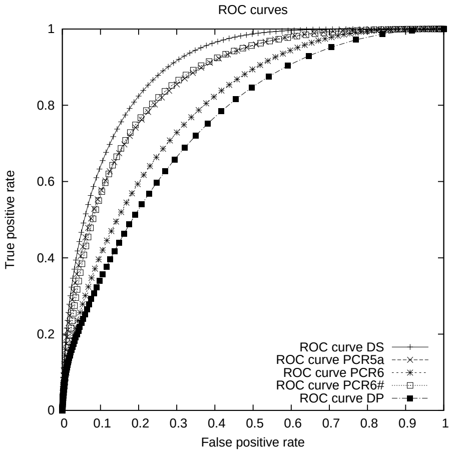

Figure 7: ROC curves for bpas $m_1(1)$ and $m_2(1)$ and m_3. Δ scale factor is 0.200

5 Related Work

Several research papers exist in literature that demonstrate the effectiveness of the Dempster-Shafer theory to combine multiple pieces of evidence and get an accurate picture of the context to be monitored and analyzed. In this section we selected the most relevant papers showing how the Dempster-Shafer theory can help in network security to spot intrusions or to detect frauds.

In ([17]) the authors present an approach for credit card fraud detection that combines different types of evidence by using the DS theory. In the proposed FDS a number of rules, like average daily / monthly spending of a customer, shipping address being different from billing address, etc., are used to analyze the deviation of each incoming transaction from the normal profile of the cardholder by assigning initial beliefs to it. The initial belief values are combined in order to obtain an overall belief by applying the DS theory. The overall belief is

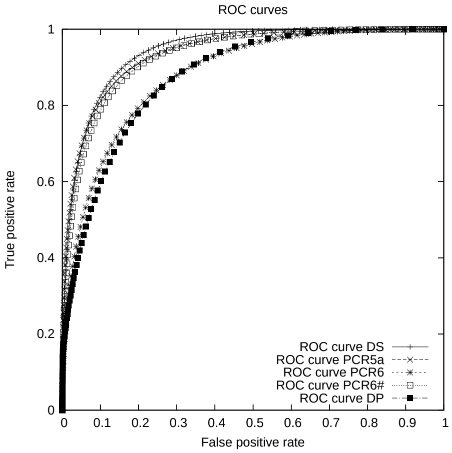

Figure 8: ROC curves for bpas $m_1(1)$ and $m_2(2)$ and m_3. Δ scale factor is 0.200

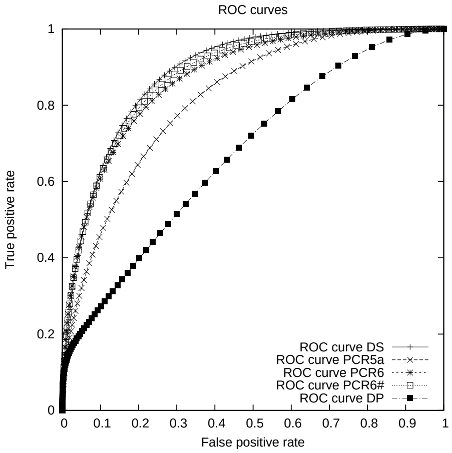

Figure 9: ROC curves for bpas $m_1(2)$ and $m_2(0)$ and m_3. Δ scale factor is 0.200

further strengthened or weakened according to its similarity with fraudulent or genuine transaction history using Bayesian learning. The authors demonstrate the effectiveness of the proposed FDS by testing it with large scale data. Due to unavailability of real life credit card data or benchmark data set for testing, they developed a simulator to generate synthetic transactions that represent the behavior of genuine cardholders as well as that of fraudsters.
In ([24]) authors propose the use of DS model to perform intrusion detection on the DARPA dataset of network related attacks. The work points out the limits of DS model in case of conflicting information sources and proposes to use a context-dependent operator. That operator changes the combination rule (conjuntive, disjuntive or avarage) based on the degree of conflict among the sources.
In ([3]), the authors investigate the use of Dempster-Shafer evidence theory for intrusion detection in ad-hoc networks. A common problem in distributed intrusion detection is how to combine observational data from multiple nodes that can vary in their reliability or

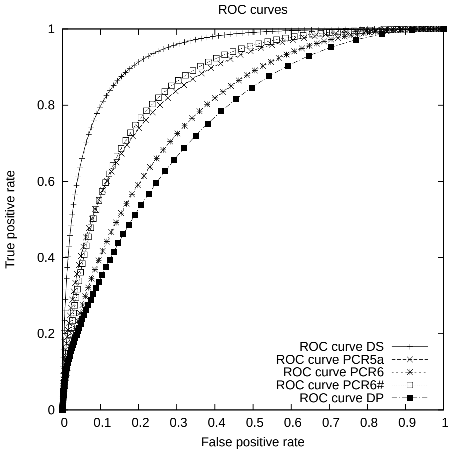

Figure 10: ROC curves for bpas $m_1(2)$ and $m_2(1)$ and m_3. Δ scale factor is 0.200

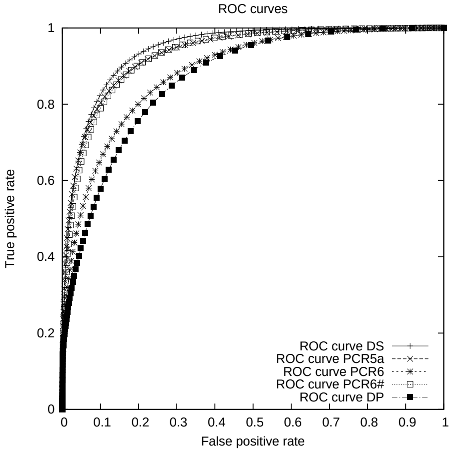

Figure 11: ROC curves for bpas $m_1(2)$ and $m_2(2)$ and m_3. Δ scale factor is 0.200

trustworthiness. Other approaches have used simplistic combination techniques such as averaging or majority voting. In this study the authors demonstrate that the DS theory is well suited to this type of problem. First, it reflects uncertainty or a lack of complete information, and second, Dempster rule for combination gives a convenient numerical procedure for fusing together multiple pieces of data.

In ([10]) a two-step approach is proposed to accurately detect shilling behavior for online auction systems. In the first step, the authors adopt a model checking method to detect suspicious shilling behaviors in real-time. To verify the detection results from the first step and to reduce the number of possible false positives, in the second step, knowledge obtained in the first step is combined and the combination is carried out using the Dempster-Shafer theory. This two-step process for shill inference produces a shilling score that can assist an auction house with trust judgment for each shill suspect. To demonstrate the feasibility of the proposed approach, the authors provide a case study using real eBay auction data. The results show that using the DS theory to combine multiple sources of evidence of shilling behaviour the approach can reduce the number of false positives that would be generated from a single source of evidence.

In ([1]) the authors used the DS theory to develop an algorithm for protecting Wireless Sensor Networks (WSNs) from internal attacks. In the reference scenario a number of sensors in the WSN are nodes, for which the observations are assumed independent of each other. The Dempster-Shafer evidence combination rule provides a means to combine these observations. The study conducted by the authors assumes that the neighbor nodes with one hop will observe the data of the suspected internal attacker. In these observations, without loss of generality, the physical parameter (temperature) and transmission behavior (packet dropping rate) for each sensor are considered as independent events. The proposed algorithm observes neighbor nodes in the WSN and uses the two parameters to make judgments for the behavior based on the DS theory. The DS theory considers the observed data as a hypothesis. If there is uncertainty about which hypothesis the data fits best, the DS theory makes it possible to model several single pieces of evidence within the relations of multiple hypotheses. Using this method the system does not need any a prior knowledge of the pre-classified training data of the nodes in a WSN.

In ([27]) a DS theory-based approach is proposed to handle the uncertainty due to the large rate of false positives in the sensors used by Intrusion Detection Systems. This approach relies on an algorithm that performs DS belief computation on an IDS alert correlation graph, thus allowing to determine a belief score for a given hypothesis, e.g. a specific machine is compromised. The belief strength can be used to sort incident-related hypotheses and prioritize further analysis of the hypotheses and the associated evidence by a human analyst. The authors have implemented the proposed approach for the open-source IDS Snort and evaluated its effectiveness on a number of data sets as well on a production network.

In ([13]) two network probes collecting traffic data are used as sensors that feed an Intrusion Detection System based on the Dempster-Shafer theory. This IDS uses the combination rule to correlate the collected data in order to detect DDoS attacks.

In ([25]) the authors propose an algorithm based on the exponentially weighted Dempster-Shafer Theory of Evidence to improve and assess alert accuracy. In order to test the proposed approach off-line experiments have been performed by using two DARPA 2000 DDoS evaluation data sets. The experimental results demonstrated that the proposed alert fusion algorithm based on an extended version of the Dempster-Shafer theory provides better performance than an alert correlation engine relying on Hidden Colored Petri-Net (HCPN).

In ([14]) the Theory of Evidence by Dezert and Smarandache (DSmT) is used in the threat assessment domain. DSmT distinguishes two operations: combination and conditioning for fusion of uncertain information and integration of uncertain pieces of information with confirmed i.e. certain evidence respectively. However, each of these operations has its drawbacks and, therefore, another type of fusion rules, called relative conditioning, has been proposed. In this kind of rules the predominance of the condition over the uncertain evidence is stated explicitly, while the trust in the conditioning hypothesis is not absolute by definition. In this paper two of these rules are presented as possible solution of the multi-level conditioning in threat assessment problem.

6 Conclusions

The Mobile Money Transfer industry is rapidly expanding around the world and this growth is particularly fast in less developed countries, where people see the MMT service as a valid and appealing alternative to the traditional and poorly disseminated banking agencies. Unfortunately, as the number of people using this service grows, the MMT infrastructure is exposed to increasingly sophisticated frauds. This paper addressed a challenging security misuse case concerning MMT services. This misuse case is called Account Takeover and it takes place when a fraudster performs money transfers by using the mobile phone stolen to a legitimate service customer. The choice of that misuse case does not represent a limitation for our fraud detection approach. Different misuse cases would require other detectors or event probes in addition or in place to those used in this work. In order to spot these illicit financial transactions the fraud detection system is required to collect, process, and correlate a massive amount of data regarding the operations performed by the service customers. Data fusion techniques can definitively help improve the performance and effectiveness of the correlation process supporting the detection task. In this paper we presented a component-based Fraud Detection System that implements data fusion algorithms deriving from the Dempster-Shafer theory of evidence. These algorithms use combination rules and procedures facing the problem of conflicting degree of belief affecting the DS theory. An extensive experimental campaign has been conducted in order to test and validate these data fusion algorithms in the MMT account takeover detection scenario.

References

[1] Muhammad Ahmed, Xu Huang, Dharmendra Sharma, and Li Shutao. Wireless sensor network internal attacker identification with multiple evidence by dempster-shafer theory. In *Algorithms and Architectures for Parallel Processing*, pages 255–263. Springer, 2012.

[2] CCK. Quarterly sector statistics report. *Communications Commission of Kenya*, 2012.

[3] Thomas M Chen and Varadharajan Venkataramanan. Dempster-shafer theory for intrusion detection in ad hoc networks. *Internet Computing, IEEE*, 9(6):35–41, 2005.

[4] Luigi Coppolino, Salvatore DAntonio, Valerio Formicola, Carmine Massei, and Luigi Romano. Use of the dempster-shafer theory for fraud detection: The mobile money transfer case study. 570:465–474, 2015.

[5] Frédéric Dambreville. Generic implementation of fusion rules based on referee function. In *Proc. of Workshop on the theory of belief functions*, 2010.

[6] Frédéric Dambreville. Release zero. 0.1 of package refereetoolbox. *arXiv preprint arXiv:1003.2641*, 2010.

[7] Frédéric Dambreville and Délégation Générale pour lArmement. Definition of evidence fusion rules based on referee functions. *Advances and Applications of DSmT for Information Fusion, Vol. 3: Collected Works*, 3:185, 2009.

[8] Arthur P Dempster. A generalization of bayesian inference. Technical report, DTIC Document, 1967.

[9] Florentin Smarandache Jean Dezert. Proportional conflict redistribution rules for information fusion. *Advances and Applications of DSmT for Information Fusion (Collected works), second volume: Collected Works*, 2:3, 2006.

[10] Fei Dong, Sol M Shatz, and Haiping Xu. Inference of online auction shills using dempster-shafer theory. In *Information Technology: New Generations, 2009. ITNG'09. Sixth International Conference on*, pages 908–914. IEEE, 2009.

[11] Didier Dubois and Henri Prade. On the relevance of non-standard theories of uncertainty in modeling and pooling expert opinions. *Reliability Engineering & System Safety*, 36(2):95–107, 1992.

[12] Chrystel Gaber, Baptiste Hemery, Mohammed Achemlal, Marc Pasquet, and Pascal Urien. Synthetic logs generator for fraud detection in mobile transfer services. In *Collaboration Technologies and Systems (CTS), 2013 International Conference on*, pages 174–179. IEEE, 2013.

[13] Wei Hu, Jianhua Li, and Qiang Gao. Intrusion detection engine based on dempster-shafer's theory of evidence. In *Communications, Circuits and Systems Proceedings, 2006 International Conference on*, volume 3, pages 1627–1631. IEEE, 2006.

[14] Ksawery Aleksander Krenc and Florentin Smarandache. Application of new absolute and relative conditioning rules in threat assessment. In *FUSION*, pages 109–114, 2013.

[15] Arnaud Martin and Christophe Osswald. A new generalization of the proportional conflict redistribution rule stable in terms of decision. *Advances and Applications of DSmT for Information Fusion (Collected works), second volume: Collected Works*, 2:69, 2006.

[16] Cynthia Merritt. Mobile money transfer services: The next phase in the evolution of person-to-person payments. *Journal of Payments Strategy & Systems*, 5(2), 2011.

[17] Suvasini Panigrahi, Amlan Kundu, Shamik Sural, and Arun K Majumdar. Use of dempster-shafer theory and bayesian inferencing for fraud detection in mobile communication networks. In *Information Security and Privacy*, pages 446–460. Springer, 2007.

[18] C. Penicaud. State of the industry: Results from the 2012 global mobile money adoption survey. In *GSMA Mobile Money for the Unbanked*, 2012.

[19] Krispijn A. Scholte and Wilbert van Norden. Applying the pcr6 rule of combination in real time classification systems. pages 1665–1672, 2009.

[20] Kari Sentz and Scott Ferson. *Combination of evidence in Dempster-Shafer theory*, volume 4015. Citeseer, 2002.

[21] F Smarandache and J Dezert. Applications and advances of dsmt for information fusion, 2004.

[22] Rajendra P. Srivastava. The dempster-shafer theory: An introduction and fraud risk assessment illustration. 2011.

[23] A. Tchamova and J. Dezert. On the behavior of dempster's rule of combination and the foundations of dempster-shafer theory. pages 108–113, Sept 2012.

[24] C. Thomas and N. Balakrishnan. Modified evidence theory for performance enhancement of intrusion detection systems. In *Information Fusion, 2008 11th International Conference on*, pages 1–8, June 2008.

[25] Dong Yu and Deborah Frincke. Alert confidence fusion in intrusion detection systems with extended dempster-shafer theory. In *Proceedings of the 43rd annual Southeast regional conference-Volume 2*, pages 142–147. ACM, 2005.

[26] L.A. Zadeh. On the validity of dempster's rule of combination of evidence. *Memo M, 79:24*, 1979.

[27] Loai Zomlot, Sathya Chandran Sundaramurthy, Kui Luo, Xinming Ou, and S Raj Rajagopalan. Prioritizing intrusion analysis using dempster-shafer theory. In *Proceedings of the 4th ACM workshop on Security and artificial intelligence*, pages 59–70. ACM, 2011.