Verhaltensanalyse zur Erkennung von Missbrauch mobiler Geldtransferdienste*

Roland Rieke[1,2], Maria Zhdanova[2], Jürgen Repp[2], Romain Giot[3,4], Chrystel Gaber[5]

[1]Philipps-Universität Marburg, Marburg, Germany
[2]Fraunhofer SIT, Darmstadt, Germany
[3]Université Bordeaux, LaBRI, UMR 5800, F-33400 Talence, France
[4]CNRS, LaBRI, UMR 5800, F-33400 Talence, France
[5]Orange, Caen, France
{roland.rieke, maria.zhdanova, juergen.repp}@sit.fraunhofer.de,
romain.giot@u-bordeaux1.fr, chrystel.gaber@orange.com

Abstract: Die fortlaufende Überwachung von Transaktionen auf Geldwäscheverdacht ist Finanzinstituten in Deutschland und anderen Ländern vorgeschrieben. Smurfing ist eine Form der Geldwäsche, bei der durch den Transfer vieler kleiner Geldbeträge auf unterschiedlichen Wegen mit der Hilfe von Strohmännern ein hoher Geldbetrag unauffällig transferiert werden soll. In dieser Arbeit betrachten wir das Smurfing-Risiko im Rahmen mobiler Geldtransferdienste. Insbesondere beschreiben wir eine Methode zur vorbeugenden Sicherheitsanalyse zur Laufzeit, welche das Prozessverhalten in einem Geldtransfer-Service in Bezug auf Transaktionen beobachtet und versucht, es mit dem erwarteten Verhalten zu vergleichen, welches durch ein Prozessmodell vorgegeben ist. Wir analysieren Abweichungen von der vorgegebenen Verhaltensspezifikation auf Anomalien, die einen möglichen Missbrauch des Finanzdienstes durch Geldwäscheaktivitäten anzeigen. Wir bewerten die Anwendbarkeit der Vorgehensweise und beschreiben Messungen der Rechen- und Erkennungsleistung eines prototypischen Werkzeugs basierend auf realen und simulierten Betriebsprotokollen. Das Ziel der Experimente ist es, basierend auf Eigenschaften des realen Finanzdienstes, Missbrauchsmuster in synthetisiertem Prozessverhalten mit eingefügten Geldwäscheaktivitäten zu erkennen.

1 Einleitung

Dienste im Bereich mobiler Geldtransfer (MGT) sind ein wachsendes Marktsegment, insbesondere in Entwicklungsländern, in denen das Netz der Bankfilialen nicht so dicht ist. So hatte beispielsweise M-Pesa, welches 2007 in Kenia eingeführt wurde, im Dezember 2011 bereits etwa 19 Millionen Teilnehmer, das entspricht 70 % aller Mobilfunkteilnehmer in Kenia [CCK12]. Orange Money ist in 10 Ländern im Einsatz und wird von rund 14 % der Mobilfunkteilnehmer dieser Länder genutzt [Ora12]. Diese Dienste ermöglichen Geschäfte

*Diese Arbeit ist eine gekürzte deutsche Version des Artikels: "Fraud Detection in Mobile Payment Utilizing Process Behavior Analysis", ARES 2013 [RZR+13]

mit elektronischem Geld, welches auch als mMoney bezeichnet wird. Die Benutzer können Bargeld über Distributoren in mMoney konvertieren, um damit Waren bei Händlern zu kaufen, Rechnungen zu bezahlen oder es an andere Nutzer des Dienstes übertragen zu lassen. Wie jeder andere Geldtransferdienst, ist ein MGT-Dienst der Gefahr ausgesetzt, zur Geldwäsche (GW) missbraucht zu werden. Dabei wird eine Verschleierung illegal erwirtschafteter Mittel genutzt, um sie in den legalen Wirtschaftskreislauf einzuschleusen. Es liegt also in der Verantwortung der MGT-Dienstleister, verdächtige GW-Aktivitäten zu erkennen und an die Behörden zu melden. Daraus ergibt sich die Bedeutung von Software-Werkzeugen, wie zum Beispiel von Security Information and Event Management (SIEM)-Systemen, welche Kundendaten zur Laufzeit analysieren und verdächtige Transaktionen erkennen.

In dieser Arbeit wird ein Werkzeug namens Predictive Security Analyzer (PSA) vorgestellt, welches auf einer neuartigen Methode zur prädiktiven Sicherheitsanalyse zur Laufzeit aufbaut [RS10, ER11]. Die Kernidee ist dabei, dass in Ergänzung zu existierenden Verfahren, die statistische Werte aus historischen Messungen verwenden, um Anomalien zu entdecken, hier das Wissen um den geplanten Kontrollfluss der Prozesse verwendet wird. Diese Methode ermöglicht die Auswertung von sicherheitsrelevanten Ereignissen und deren Interpretation in Bezug auf: (1) den spezifizierten Kontrollfluss der Prozesse und (2) die erforderlichen Sicherheitseigenschaften. Bezüglich (1), werden Abweichungen des beobachteten Prozessverhaltens von der vorgegebenen Spezifikation identifiziert. Solche Abweichungen können durch Änderungen in der Prozessausführung, Probleme bei der Messung (z.B., verlorene Ereignisse) oder Anomalien durch Interventionen eines Angreifers verursacht werden. Bezüglich (2) wird eine kontinuierliche Überwachung der für den Prozess festgelegten Sicherheitseigenschaften durchgeführt, um mögliche Sicherheitsverletzungen zu erkennen und vorherzusagen. In diesem Beitrag evaluieren wir die Anwendbarkeit des Ansatzes, um Missbrauchsmuster in den Ereignisströmen der MGT-Transaktionen zu identifizieren, die auf GW-Aktivitäten hinweisen. Dies ist, soweit wir wissen, die erste Anwendung dieser Sicherheitsanalysemethode im MGT-Kontext. Insbesondere zeigen wir, dass der PSA in der Lage ist, (1) einen Ereignisstrom aus einem MGT-System in Echtzeit zu verarbeiten und (2) GW-Warnungen bei betrügerischen Transaktionen zu generieren. Wir beschreiben Ergebnisse von Experimenten mit realen und simulierten Transaktionsprotokollen für verschiedene GW-Szenarien und ermitteln die Sensitivität und Leistungsfähigkeit des PSA.

Das Arbeit ist wie folgt gegliedert: Kapitel 2 stellt das MGT-Szenario und einen Missbrauchsfall im Zusammenhang mit GW vor und Kapitel 3 gibt einen Überblick über den PSA und dessen Anwendung im Experiment. Kapitel 4 beschreibt den Versuchsaufbau und Kapitel 5 präsentiert die experimentellen Ergebnisse. Kapitel 6 gibt einen Überblick über verwandte Arbeiten und Kapitel 7 gibt ein Fazit dieser Arbeit.

2 Missbrauchserkennung in einem System für mobilen Geldtransfer

Diese Arbeit basiert auf einem Anwendungsfall, der in [AGG+11] im Detail beschrieben ist. Dieser Abschnitt beschreibt die wichtigsten Punkte, um den Anwendungsfall zu verstehen. MGT-Systeme sind Systeme, in denen verschiedene Arten von Finanztransaktionen (zum

Beispiel das Ein- und Auszahlen von Bargeld, nationale und internationale Überweisungen, Rechnungszahlungen, etc.) unter Verwendung von mMoney durchgeführt werden. Ein Mobilfunkbetreiber (MFB) emittiert mMoney in Partnerschaft mit einer Privatbank und sendet regelmäßig Complianceberichte an die Zentralbank, die für die Geldpolitik des Landes verantwortlich ist. Das emittierte mMoney kann nur von MFB-Kunden, die den MGT abonniert haben, verwendet werden. Die Teilnehmer sind Endkunden, Dienstleister, Händler und Geschäfte. Diese besitzen ein Prepaid-Konto auf einer MGT-Plattform, das über das Netz des MFB erreichbar ist. Für den Zugriff auf ihre mMoney-Konten verwenden die Endkunden eine dazu passende Anwendung auf ihren mobilen Geräten.

Wir konzentrieren uns hier auf einen Fall von Missbrauch im Zusammenhang mit GW über einen MGT-Dienst wie in Abbildung 1 dargestellt. Die farbigen Pfeile zeigen die regulären Überweisungen, leere Pfeile bezeichnen die betrügerischen. Ein böswilliger Be-

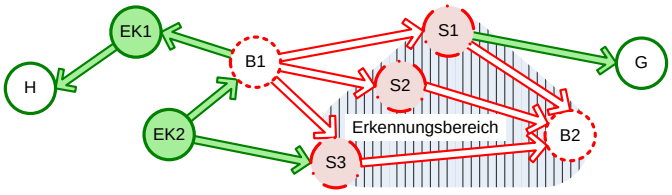

Abbildung 1: MGT Benutzer im Geldwäsche-Szenario: Endkunde (EK), Betrüger (B), Strohmann (S), Händler (H), Geschäft (G)

nutzer (Betrüger) überträgt kleine Geldbeträge auf mehrere Strohmänner. Strohmänner erhalten mMoney von einem Betrüger und überweisen einen hohen Prozentsatz dieses mMoney an einen anderen Betrüger. Jeder Strohmann kann einen kleinen Prozentsatz der Überweisung für sich behalten. Der jeweils letzte Strohmann in einer Kette initiiert den finalen Geldtransfer an einen zweiten betrügerischen Benutzer. Die in dieser Arbeit durchgeführte GW-Analyse basiert auf den in der gestreift markierten Zone dargestellten finalen Geldtransfers. In dieser eingeschränkten Sicht transferiert also ein betrügerischer Benutzer mMoney an einen zweiten betrügerischen Benutzer aber es gibt keine direkten Übertragungsspuren zwischen diesen im MGT-System.

Es gibt eine Vielzahl von GW-Verfahren [FIN12, Int12]. In dieser Arbeit betrachten wir ein GW-Schema mit folgenden Annahmen: (i) es gibt nur einen Strohmann in der Kette der Strohmänner; (ii) der Betrag einer betrügerischen Transaktion ist viel kleiner als der Durchschnitt in diesem Dienst; (iii) Strohmänner führen, außer den betrügerischen Aktionen, auch reguläre mMoney-Überweisungen durch. Diese Annahmen schränken den vorgeschlagenen Ansatz zur Betrugserkennung nicht ein, solange man in der Lage ist, einen Prozessablauf zu einem anderen gewählten GW-Schema zu spezifizieren. Weiterhin hat die Analyse der realen Betriebsprotokolle, die uns vom Betreiber des Dienstes für diese Untersuchung zur Verfügnung gestellt wurden, gezeigt, dass es bei MGT-Benutzern normalerweise keine plötzlichen Änderungen in der Höhe der Transaktionsbeträge gibt.

3 Prädiktive Sicherheitsanalyse zur Laufzeit

Prädiktive Sicherheitsanalyse nutzt eine formalisierte, ausführbare Prozessspezifikation zu einer Berechnung des geplanten Prozessverhaltens im jeweiligen Zustand des überwachten Systems. Während der Laufzeit wird nun das tatsächliche Prozessverhalten aus dem Ereignisstrom des MGT-Systems gemessen und mit dem geplanten Verhalten verglichen, um Anomalien festzustellen. Desweiteren können Sicherheitsanforderungen spezifiziert werden, welche zur Laufzeit am gemessenen Prozessverhalten überprüft werden. Darüber hinaus wird das Wissen über das erwartete Verhalten – aus dem Prozessmodell – verwendet, um Fehler in der nahen Zukunft vorherzusagen. Diese prädiktive Sicherheitsüberwachung ermöglicht es, negativen künftigen Aktionen proaktiv zu begegnen.

Der PSA unterstützt den vollständigen Zyklus dieser ereignisbasierten Sicherheitsanalyse. Die in der MGT-Anwendung verwendeten Funktionen werden im Folgenden beschrieben.

Zunächst extrahiert der PSA Ereignisse aus einem Ereignisstrom oder liest Ereignisse aus einer Datenbank, entsprechend einem vordefinierten Ereignisschema. Basierend auf dem Ereignisschema werden abstrakte Ereignisse definiert, um für die Sicherheitsanalyse nicht relevante Informationen herauszufiltern.

Die vom PSA durchgeführten Simulationen basieren auf formalen Prozessrepräsentationen, welche mittels Asynchroner Produktautomaten (APA) [RRZE14] spezifiziert werden. Ein APA kann als eine Familie von Elementarautomaten angesehen werden, die über gemeinsame Zustandskomponenten kommunizieren. Informale Spezifikationen der Prozesse, die in der Ereignisgesteuerte Prozesskette (EPK)-Notation [KNS92] vorliegen, müssen also entsprechend formalisiert werden, um eine Sicherheitsanalyse durchführen zu können. Die EPK-Notation ist für Anwender, die nicht mit formalen Spezifikationsmethoden vertraut sind, leicht verständlich. Der PSA unterstützt den Benutzer beim Erstellen eines EPK sowie dessen Transformation in ein operationales formales Modell. Existierende EPK-Modelle können unter Verwendung von archivierten Ereignisprotokollen, die als Eingabe benutzt werden, oder auch interaktiv direkt zur Laufzeit angepasst werden. Auf diese Weise wird ein *Uncertainty Management* realisiert, dass eine halbautomatische Anpassung des Prozessmodells, entsprechend den aktuellen Kontextbedingungen, durchführt. Uncertainty-Situationen können insbesondere während der Synchronisierung des Zustands einer laufenden Prozessinstanz mit dem Zustand des Prozessmodells auftreten, wenn das Modell nicht hinreichend exakt oder veraltet ist, unerwartete Ereignisse auftreten oder erwartete Ereignisse nicht empfangen werden. Neben APA-Modellen können Petri Net Markup Language (PNML)-Spezifikationen [WK03], die von Process-Discovery-Werkzeugen generiert werden (z.B., ProM [MMWvdA11]), importiert werden. Für alle Spezifikationsmethoden wird die Berechnung der nächsten Systemzustände auf Modellebene unterstützt.

Sicherheitseigenschaften, die der zu untersuchende Prozess erfüllen soll, werden in Form von Monitorautomaten [RRZE14] spezifiziert. Wird ein Ereignis aus dem überwachten System erkannt, das zu einem Zustandsübergang in einem solchen Monitorautomaten passt, so wird der aktuelle Zustand dieses Automaten entsprechend geändert. Falls ein als kritisch eingestufter Zustand erreicht wird, so wird ein entsprechender Alarm generiert. Falls der PSA in dem Vorhersagehorizont des Prozessverhaltens einen möglichen Zustandsübergang

findet, der zu einem kritischen Zustand in einem Monitorautomaten führt, so wird ein "prädiktiver" Alarm generiert. Der PSA unterstützt die Visualisierung des aktuellen Prozesszustandes und des aktuellen Sicherheitsstatus in Form des Zustands der betroffenen Monitorautomaten.

Der PSA wurde in der vorliegenden Arbeit in zwei Phasen benutzt, einer *Lernphase* und einer *Anomalieerkennungsphase*. In der initialen Lernphase wird das Normalverhalten un-

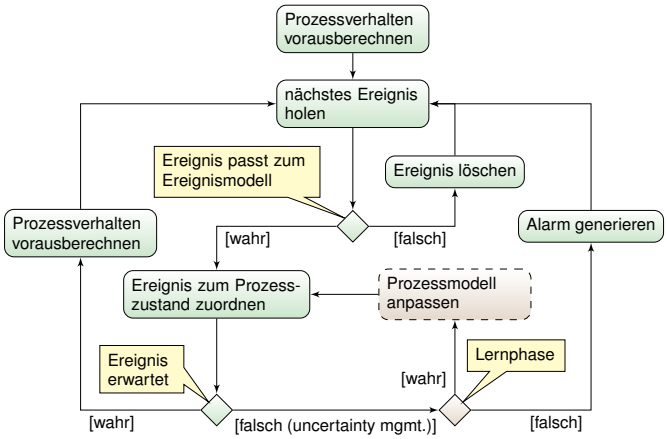

Abbildung 2: Laufzeitverhalten des PSA

ter Verwendung existierender Log-Dateien, die nur korrektes Verhalten enthalten, erlernt. Für diesen Zweck wird eine Abbildung, bei der eine Klassifikation abhängig von der transferierten Geldmenge definiert wird (siehe Tabelle 1), sowie die Reihenfolge der abstrakten Ereignisse aus dem Transaktionsprotokoll verwendet. Beispielweise wird ein Transaktionswert zwischen 500 und 1000 auf den abstrakten Wert $medium$ abgebildet. Diese empirisch erzeugte Abbildung, die auf Transaktionsprotokolle aus dem realen Betrieb des Dienstes angewandt wird, kann für spezifische Trainingsmengen des MGT unterschiedlich sein. Abbildung 2 zeigt eine Übersicht zu den Schritten der Ereignisverarbeitung des PSA in der Lernphase. Die durch eine gestrichelte Linie gekennzeichnete Aktion "Prozessmodell anpassen" wird halbautomatisch durchgeführt und erfordert eine Benutzerinteraktion.

In der Anomalieerkennungsphase werden Abweichungen vom charakteristischen Normalverhalten erkannt und entsprechende Alarme generiert. In unserer experimentellen Testumgebung wurde ein synthetisches Prozessverhalten durch einen Simulator erzeugt, der die Testdaten, basierend auf Eigenschaften von realen Transaktionsdaten, generiert. In der Anomalieerkennungsphase wird die Aktion "Prozessmodell anpassen" in Abbildung 2 nicht verwendet. Stattdessen erfolgt die automatische Generierung von Alarmen durch die Uncertainty-Management-Komponenten des PSA. Um die Anzahl der Fehlalarme zu reduzieren, kann das Modell des Normalverhaltens mit Hilfe von Realdaten, in denen bereits Annotationen für verdächtige Aktionen vorliegen, verbessert werden. Alternativ können entsprechende Filter für die durch den PSA generierten Alarme verwendet werden.

4 Versuchsanordnung

Der PSA wird im nicht-interaktiven Modus verwendet, so dass beim Empfang von unerwarteten Ereignissen automatisch Alarme generiert werden.

Der Transaktionsbetrag ist eine Variable aus $\mathbb{R}$, was eine Diskretisierung erfordert, um eine berechenbare Abstraktion des Verhaltens zu erhalten. Daher wurden empirisch verschiedene Transaktionsbetragsklassen definiert (siehe Tabelle 1).

Tabelle 1: Abbildung zur Diskretisierung des Transaktionsbetrags

Klasse	winzig	sehr klein	klein	normal	mittel	groß	sehr groß	riesig
Betrag	≤ 5	≤ 50	≤ 200	≤ 500	≤ 1000	≤ 2000	≤ 5000	sonst

Prozess Definition: Um die PSA-Analyse durchzuführen, muss der MGT-Prozess mittels EPK definiert werden. Der PSA unterstützt die Möglichkeit mehrere parallel laufende Benutzerprozesse, deren Instanzen das gleiche PSA-Modell zugrunde liegt, mittels dieses Modells zu untersuchen. Hier wurde das allgemeine Verhalten von allen Benutzern verwendet. Da jeder Benutzer völlig frei in der Art der Nutzung des MGT Systems ist (z.B. Auswahl des Betrags, Häufigkeit von Transaktionen, Interessengemeinschaften, etc.), war es eine Herausforderung die entsprechenden Transaktionsabläufe zu definieren. Aus diesem Grund wurde ein Prozess spezifiziert, der aus einem Missbrauchsszenario abgeleitet wurde: Jeder Prozesszustand bezieht sich auf einen bestimmten Transaktionsbetrag, wie in der Diskretisierung definiert. Zustandsübergänge werden durch die korrespondierenden abstrakten Ereignisse angestoßen. Für jeden Zustand sind nur Übergänge in der gleichen Transaktionsbetragsklasse oder zu den benachbarten Transaktionsbetragsklassen autorisiert; alle anderen Übergänge werden als bösartig betrachtet und erzeugen einen Alarm. Der initiale Zustand erlaubt jeden Übergang.

Operationale Logs: Im realen MGT-System existieren verschiedene Arten von Logs (Zugriff, Transfer, etc.). Für unsere Analyse wurden aber nur die Logs von Transaktionen verwendet. Ein Log-Datensatz enthält den Betrag, den Sender und den Empfänger der Transaktion, den Typ des Senders und des Empfängers sowie weitere systemspezifische Felder. Ereignisse, die Log-Einträge generieren, werden durch das Verhalten des Benutzers ausgelöst sobald beliebige Benutzer des Systems Transaktionen durchführen. Im Rahmen des MASSIF-Projektes wurde von Orange für unsere Analyse ein anonymisiertes Log zur Verfügung gestellt, welches 4,5 Millionen Transaktionen enthält, die in einem Zeitraum von 9 Monaten erzeugt wurden. Diese Realdaten wurden benutzt, um eine Echtzeitanalyse des MGT-Datenstroms mittels des PSA durchzuführen. Da keine Gewissheit bzgl. der Ereignisse (d.h., betrügerisch oder normal) besteht, ist eine direkte Betrugserkennung nicht möglich. Um die Erkennungsrate zu untersuchen, wurden daher Simulations-Logs verwendet.

Simulierte Ereignisse: Um Ereignislogs mit sicheren Fehlerraten zu produzieren, wurde ein Missbrauchsfall in einem Simulator [GHA+13] implementiert. Der Simulator modelliert einzelne Trajektorien als eine Folge von Transaktionen mit Beträgen und Zeitintervallen nach Normalverteilung. Die generierten Simulationsdaten basieren auf Eigenschaften, die an realen Beispielen ermittelt wurden und enthalten unterschiedliche Benutzerkategorien aus dem GW-Szenario (siehe Abbildung 1). *Endkunden* führen regelmäßige Transaktionen (Mittelwert 4000, mit einer Standardabweichung von 500), Abhebungen und Einzahlungen durch. *Betrüger* nutzen den Dienst zur Geldwäsche. *Strohmänner* empfangen mMoney ($20 \leq$ Betrag ≤ 100) von einem Betrüger und transferieren den Betrag zu einem weiteren Betrüger unter Einbehaltung von 10% des Betrags. Betrüger und Strohmänner führen ebenfalls regelmäßige Transaktionen durch. *Geschäfte* akzeptieren mMoney als Zahlungsmittel. *Händler* ermöglichen Endkunden, mMoney in Bargeld umzutauschen und umgekehrt. Das GW-Szenario wurde mit den folgenden Parametern konfiguriert:

S1 keine GW: 50 Endkunden, 8 Geschäfte, 4 Händler. Dieser Datensatz wurde verwendet, um zu überprüfen, ob ehrliche Endkunden als Betrüger eingestuft werden.

S2 GW: 50 Endkunden, 5 Strohmänner, 8 Geschäfte, 4 Händler. Dieser Datensatz diente zur Verifikation der Erkennung von Betrugsfällen durch den PSA.

S3 GW mit mehr Beteiligten: 500 Endkunden, 10 Strohmänner, 16 Geschäfte, 4 Händler. Hierbei wurde die Erkennungsrate bei einem kleineren Anteil an betrügerischen Transaktionen bestimmt.

Evaluierungsmetrik: Die vorliegende Analyse betrachtet die Leistungsfähigkeit bzgl. der Laufzeit, sowie der Erkennung von Betrugsversuchen durch den PSA. Es wurde untersucht, ob die Analyse, unter Realzeitbedingungen eines laufenden MGT-Systems, durchgeführt werden kann und ob ein bestimmtes Zeitintervall von der Erkennung bis zur Signalisierung eines Alarms eingehalten werden kann. Die Untersuchungen wurden auf einem Personal Computer (2 Kerne CPU mit 2.70GHz, 4Gb RAM) durchgeführt. Abschließend haben wir die Fehlerrate des PSA gemessen. Die Leistung wird in Anzahl von Ereignissen, die erfolgreich durch den PSA in einer Sekunde bearbeitet werden, gemessen. Für die Erkennungsrate werden unterschiedliche Metriken benutzt: (a) *False Positive*, nicht bösartiges Ereignis wird als bösartig erkannt; (b) *False Negative*, bösartiges Ereignis wird als nicht bösartig erkannt; (c) *True Positive*, bösartiges Ereignis wird als bösartig eingestuft; (d) *True Negative*, nicht bösartiges Ereignis wird als nicht bösartig eingestuft.

5 Experimentelle Ergebnisse

In diesem Abschnitt werden die Ergebnisse der Experimente zusammengefasst.

Rechenleistung: Reale Ereignisse der operationalen Logs des MGT-Systems wurden benutzt, um die Leistungsfähigkeit des PSA zu evaluieren. Prozessinstanzen wurden für

die Kombination aus Benutzerkennung und Transaktionstyp erzeugt. Mit den realen Logs konnte der PSA 640.000 Prozessinstanzen bearbeiten und brauchte 40 Minuten, um die zugehörigen 5,5 Millionen Ereignisse zu verarbeiten und 0,5 Millionen Alarme zu erzeugen. Eine vollständige Analyse auf der gleichen Datenmenge, die keine Alarme produzierte, benötigte 33 Minuten. Im günstigsten theoretischen Fall (keine Alarme werden generiert) kann der PSA mehr als 2.300 Ereignisse / Sek. verarbeiten, während im ungünstigsten Fall die Anzahl sich auf ca. 191 Ereignisse / Sek. reduziert. Im Mittel kann der PSA 100 Millionen Ereignisse an einem Tag auf einem Standard Computer verarbeiten.

Erkennungsrate: Um die Erkennungsrate des PSA zu bestimmen, wurden simulierte Ereignisse verwendet. Aufgrund des stochastischen Charakters der Simulation wurde die Evaluierung mehrmals durchgeführt. Alle Versuche führten zu den gleichen Ergebnissen. Abbildung 3 illustriert die Transaktionen, die von den an Betrugsversuchen Beteiligten des Szenarios S2 durchgeführt wurden (665 Transaktionen). Jeder Knoten repräsentiert einen Benutzer, jede Kante zeigt eine Transaktion, die Kanten werden mit dem Index der Transaktion in der Folge beschriftet. Die grauen Kanten zeigen True Negative, orange (gepunktete) Kanten – False Positive, grüne (dunkelgrau) Kanten bilden True Positive ab, rote (gestrichelte) Kanten – False Negative. Bösartige Transaktionen vom ersten Betrüger zu

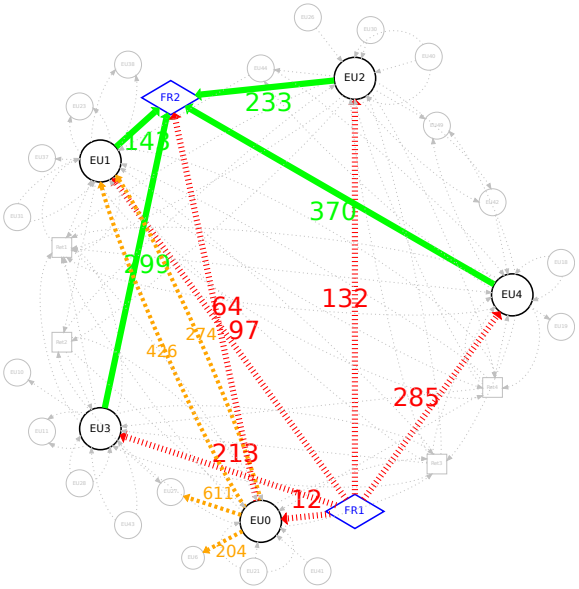

Abbildung 3: Benutzer (EU), Betrüger (FR) und Transaktionen in Szenario S2

den Strohmännern werden nicht als betrügerisch erkannt, was korrekt ist, da die Erkennung solcher Transaktionen nicht intendiert war. Transaktionen von allen Strohmännern, außer $EU0$, zu dem zweiten Betrüger werden korrekt als Betrugsversuch erkannt. Reguläre Transaktionen des Strohmanns $EU0$ werden als bösartig eingestuft und erzeugen falsche Alarme. Tabelle 2 stellt die Sensitivitäts- und Spezifitätskennzahl, die spezifischen Fehler- und Er-

kennungsraten dar. Die Werte in Klammern beziehen sich auf alle GW-Transaktionen ohne Berücksichtigung des gewählten Erkennungsbereichs (siehe Abbildung 1).

Tabelle 2: Erkennungsrate des PSA

	Spezifität	Sensitivität	False Positive Rate	False Negative Rate
S1	100%	–	0%	–
S2	≈ 99,4%	80% (40%)	≈ 0,6%	20% (60%)
S3	≈ 99,9%	90% (45%)	≈ 0,1%	10% (55%)

Da als Basis für die Untersuchung der Erkennungsrate keine realen Ereignisse dienten, kann die tatsächliche Erkennungsrate nicht verifiziert werden. Eine Reduktion der Zahl der Alarme könnte durch eine Verfeinerung der EPK mittels annotierter Realdaten (betrügerisch/nicht betrügerisch) oder mit zusätzlichen Filterkomponenten erfolgen. Das Verhalten des PSA war für alle Szenarien korrekt. Die Fehler bzgl. des Benutzers $EU0$ kommen daher, dass es die erste Transaktion dieses Benutzers ist, mit dem Folgefehler, dass alle nachfolgenden Transaktionen als betrügerisch eingestuft werden. Da das erstellte Prozessmodell nur das veränderte Benutzerverhalten betrachtet, zeigt die zeitliche Abfolge der verdächtigen Transaktionen keine Wirkung auf die Erkennungsfähigkeit des PSA.

Normalerweise wird die Evaluierung von Anomalie-Erkennungswerkzeugen mittels einer ROC-Kurve [Faw04] (spezifiziert durch verschiedene Konfigurierungsschwellwerte $\tau \in \mathbb{R}$) durchgeführt. Der PSA kann nicht mittels solch einfacher Schwellwerte konfiguriert werden. Stattdessen, ist eine komplexe Konfiguration ($\rho = (\rho_{EPC}, \rho_{mapping})$) zusammengesetzt aus einer EPK-Konfigurierung ($\rho_{EPC} \in \mathbb{E}$, $\mathbb{E}$ ist die Menge möglicher EPKs) verbunden mit dem Diskretisierungsschema für Transaktionswerte ($\rho_{mapping} \in \mathbb{M}$, $\mathbb{M}$ ist die Menge aller möglichen Abbildungsfunktionen) erforderlich. Da es schwierig ist verschiedene automatisch generierte Konfigurationen, gegeben durch eine ROC-Kurve, zu durchlaufen, erfolgte die Beschränkung auf einen Punkt.

6 Verwandte Arbeiten

In Bezug auf die Auswertung der Ansätze im Bereich des Geschäftsprozessmanagements in [vdA13], kann man die Funktionalität des PSA-Prototyps als Verfahren zur "Überprüfung der Konformität mit Ereignisdaten" einstufen. In diesem Ansatz werden ein Prozessmodell und Ereignisdaten verwendet, um Abweichungen des Laufzeitverhaltens vom erwarteten Verhalten zu erkennen. Das Interesse für diesen Aspekt des Geschäftsprozessmanagements ist in den letzten drei Jahren gewachsen [vdA13]. Ein ähnlicher Ansatz ist in [RvdA08] beschrieben, aber der Schwerpunkt liegt dort auf einer Quantifizierung von Abweichungen durch die Bildung von Metriken. Wir betrachten die Arbeit bezüglich Laufzeit-Compliance-Überprüfung für Geschäftsprozesse in [MMWvdA11] als Ergänzung zu unserer Arbeit.

Viele Data-Mining-Algorithmen wurden zur Betrugserkennung im Bankenbereich angepasst [DSF12, WLC+13, KMK10]. Filter, Entscheidungsbäume und logistische Regressionsanalyse sind die meist verwendeten Verfahren. Warum eine bestimmte Transaktion als

betrügerisch eingestuft wird, ist mit den Ergebnissen dieser Verfahren leicht zu erklären. Verfahren, die automatisiert ein Modell lernen, werden seltener angewendet. Einige Industrielösungen verwenden jedoch solche Methoden. VISA, beispielsweise implementiert neuronale Netze in deren Betrugserkennungstool RST (Real-Time Scoring) [VIS].

Es gibt mehrere Ansätze Data-Mining-Algorithmen zur Betrugserkennung mittels neuronaler Netze, SVMs, Bayesischen Netzen, Entscheidungsbäumen, angepassten Expertensystemen und Hidden-Markov-Modellen im Umfeld von Kreditkartengeschäften [BJTW11, DAP09]. Für VISA wird in [CGL+07] ein Modell eines mehrdimensionalen Datenwürfels mit separater Änderungserkennung für jede Zelle verwendet. Nach unserer Kenntnis verwenden nicht alle mobilen Zahlungsdienste automatisierte Betrugserkennungslösungen. Die Überwachung kann manuell oder auf Basis von Geschäftsregeln stattfinden. Der M-PESA-Dienst setzt die MinotaurTM Fraud-Management-Lösung basierend auf Geschäftsregeln und neuronalen Netzen ein [Neu]. Nach unserer Kenntnis gibt es keine öffentlichen Arbeiten zur Anpassung der aufgeführten Betrugserkennungsmethoden an MGT-Systeme. Deshalb fällt ein Vergleich unsere Arbeit mit bestehenden Systemen schwer.

7 Fazit

Diese Arbeit nutzt Alarme, welche durch die Uncertainty-Komponente des PSA erzeugt werden, um Aktivitäten, die auf GW hindeuten, in einem MGT-System zu erkennen. Dabei werden GW-Muster in synthetisch generiertem Prozessverhalten, welches auf Basis einer Auswertung der Eigenschaften von Ereignisströmen aus einem realen MGT-System generiert wurde, analysiert. Wir haben gezeigt, dass der PSA in der Lage ist, Betrugserkennung in einem simulierten Szenario durchzuführen. Es konnte gezeigt werden, dass diese Erkennungsleistung effizient ist, aber empfindlich gegenüber Rauschen in einer realen Umgebung reagiert. Es ist daher notwendig, die Rauschfestigkeit durch eine Verbesserung der Korrelation der erzeugten Warnungen oder durch eine spezifische Auswertung des Prozesszustands zu verbessern. Beispielweise kann man in einen kritischen Zustand gehen, wenn die gleiche Warnung mehrfach auftritt. Ergebnisse des PSA sollten von Entscheidungunterstützungs- und Reaktionssystemen ausgewertet werden, um die Sicherheitsregeln des MGT-Systems anzupassen und die betrügerischen Transaktionen automatisch zu blockieren [RCH+12]. Um die Beurteilung des Systems zu erleichtern, wäre es interessant, Methoden zu entwickeln, welche in der Lage sind, eine große Menge von EPKs automatisch zu erzeugen und damit eine breite Basis für weitere Auswertungen bereitstellen.

Danksagung

Die vorliegende Arbeit basiert auf Forschungsergebnissen des Projektes MASSIF (ID 257475), welches durch die Europäische Kommission kofinanziert wurde, sowie des Projektes ACCEPT (ID 01BY1206D), welches durch das Bundesministerium für Bildung und Forschung gefördert wird.

Literatur

[AGG+11] M. Achemlal, S. Gharout, C. Gaber, M. Llanes, E. Prieto, R. Diaz, L. Coppolino, A. Sergio, R. Cristaldi, A. Hutchison und K. Dennie. Scenario requirements. Bericht, MASSIF FP7-257475, 2011.

[BJTW11] S. Bhattacharyya, S. Jha, K. Tharakunnel und J. C. Westland. Data mining for credit card fraud: A comparative study. *Decision Support Systems*, 50, 2011.

[CCK12] CCK. Quarterly sector statistics report. Bericht, Communications Commission of Kenya, 2012.

[CGL+07] C. Curry, R. L. Grossman, D. Locke, S. Vejcik und J. Bugajski. Detecting changes in large data sets of payment card data: a case study. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '07, Seiten 1018–1022, 2007.

[DAP09] L. Delamaire, H. Abdou und J. Pointon. Credit card fraud and detection techniques: a review. *Banks and Bank systems*, 4, 2009.

[DSF12] R. Dreżewski, J. Sepielak und W. Filipkowski. System supporting money laundering detection. *Digital Investigation*, 9(1):8 – 21, 2012.

[ER11] J. Eichler und R. Rieke. Model-based Situational Security Analysis. In *Workshop on Models@run.time*, Jgg. 794, Seiten 25–36. CEUR, 2011.

[Faw04] T. Fawcett. ROC Graphs: Notes and Practical Considerations for Researchers. *Pattern Recognition Letters*, 27(8):882–891, 2004.

[FIN12] FINTRAC Typologies and Trends Reports. Money Laundering and Terrorist Financing Trends in FINTRAC Cases Disclosed Between 2007 and 2011. `http://www.fintrac-canafe.gc.ca/publications/typologies/2012-04-eng.asp#s1-1`, April 2012. Last visit on 21/05/2013.

[GHA+13] C. Gaber, B. Hemery, M. Achemlal, M. Pasquet und P. Urien. Synthetic logs generator for fraud detection in mobile transfer services. In *Proceedings of the 2013 International Conference on Collaboration Technologies and Systems (CTS2013)*, 2013.

[Int12] Internal Revenue Service (IRS). Examples of Money Laundering Investigations. Fiscal Year 2012. `http://www.irs.gov/uac/Examples-of-Money-Laundering-Investigations-Fiscal-Year-2012`, October 2012. Last visit on 21/05/2013.

[KMK10] N. A. L. Khac, S. Markos und M.-T. Kechadi. A Data Mining-Based Solution for Detecting Suspicious Money Laundering Cases in an Investment Bank. In *Advances in Databases Knowledge and Data Applications (DBKDA), 2010 Second International Conference on*, Seiten 235–240, 2010.

[KNS92] G. Keller, M. Nüttgens und A.-W. Scheer. Semantische Prozeßmodellierung auf der Grundlage "Ereignisgesteuerter Prozessketten (EPK)". *Veröffentlichungen des Instituts für Wirtschaftsinformatik (IWi), Universität des Saarlandes*, 89, 1992.

[MMWvdA11] F. M. Maggi, M. Montali, M. Westergaard und W. M. P. van der Aalst. Monitoring Business Constraints with Linear Temporal Logic: An Approach Based on Colored Automata. In *Business Process Management (BPM 2011)*, Jgg. 6896 of *LNCS*, Seiten 132–147. Springer, 2011.

[Neu] Neural technologies. MinotaurTM Fraud Detection Software - Finance Sector. http://www.neuralt.com/fraud_detection_software.html. Last visited on 23/03/2013.

[Ora12] Orange. Orange Money. http://www.orange.com/en/press/press-releases/press-releases-2012/Orange-Money-reaches-4-million-customers-and-launches-in-Jordan-and-Mauritius, June 2012. Last visit on 12/04/2013.

[RCH+12] R. Rieke, L. Coppolino, A. Hutchison, E. Prieto und C. Gaber. Security and Reliability Requirements for Advanced Security Event Management. In I. Kotenko und V. Skormin, Hrsg., *Computer Network Security*, Jgg. 7531 of *Lecture Notes in Computer Science*, Seiten 171–180. Springer Berlin Heidelberg, 2012.

[RRZE14] R. Rieke, J. Repp, M. Zhdanova und J. Eichler. Monitoring Security Compliance of Critical Processes. In *Parallel, Distributed and Network-Based Processing (PDP), 2014 22th Euromicro International Conference on*. IEEE Computer Society, 2014.

[RS10] R. Rieke und Z. Stoynova. Predictive Security Analysis for Event-Driven Processes. In *Computer Network Security*, Jgg. 6258 of *LNCS*, Seiten 321–328. Springer, 2010.

[RvdA08] A. Rozinat und W. M. van der Aalst. Conformance checking of processes based on monitoring real behavior. *Information Systems*, 33(1):64 – 95, 2008.

[RZR+13] R. Rieke, M. Zhdanova, J. Repp, R. Giot und C. Gaber. Fraud Detection in Mobile Payment Utilizing Process Behavior Analysis. In *Proceedings of 2013 International Conference on Availability, Reliability and Security, ARES 2013*, Seiten 662–669. IEEE Computer Society, 2013.

[vdA13] W. M. P. van der Aalst. Business Process Management: A Comprehensive Survey. *ISRN Software Engineering*, Seite 37, 2013.

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

[WK03] M. Weber und E. Kindler. The Petri Net Markup Language. In *Petri Net Technology for Communication-Based Systems*, Jgg. 2472 of *LNCS*, Seiten 124–144. Springer, 2003.

[WLC+13] W. Wei, J. Li, L. Cao, Y. Ou und J. Chen. Effective detection of sophisticated online banking fraud on extremely imbalanced data. *World Wide Web*, 16(4):449–475, 2013.