Analyse der Verkettbarkeit in nutzergesteuerten Identitätsmanagementsystemen

Dogan Kesdogan
kesdogan@acm.org

Vinh Pham*
vinh.pham@gmx.net

Lexi Pimenidis†
pimenidis@nets.rwth-aachen.de

Abstract:
Nutzergesteuerte Identitätsmanagementsysteme haben als Ziel die Verkettbarkeit zwischen den verschiedenen digitalen Identitäten eines Benutzers zu verhindern. Wir führen theoretische und experimentelle Untersuchungen zu dem folgenden Informationsverlustproblem durch: Seien konsistente Beobachtungen eines starken Identitätsmanagementsystems gegeben (z.B. *Idemix*), in dem Benutzer Credentials anfordern und vorzeigen. Geben diese Beobachtungen genügend Informationen, um die Verkettungen der digitalen Identitäten der Benutzer aufzudecken?

Wir werden zeigen, dass die Pseudonyme der Benutzer theoretisch verkettbar sind und dass es ein NP-vollständiges Problem darstellt. Des Weiteren evaluieren wir praktische Instanzen des Problems und zeigen, dass die digitalen Pseudonyme eindeutig verkettbar sind, trotz vollständiger Anonymität auf der Netzwerkebene und der Benutzung eines Identitätsmanagementsystems.

1 Einleitung

Eine *digitale Identität* ist eine Ansammlung von Informationen über eine Person, die eine Organisation für verschiedene Zwecke sammelt, z.B. für Marketing. Sie besteht für gewöhnlich aus einer Menge von persönlichen Attributen wie Hobby, Bildungsgrad oder Lieblingsbücher, die zusammen unter einer Kennung gespeichert werden. Diese Kennung kann z.B. der Name der Person sein, ein Benutzername, eine E-Mail-Adresse, ein Cookie oder ähnliches sein [DD006, Ray01].

Das Sammeln von Benutzerdaten zur Generierung einer digitalen Identität steht nicht immer in Einklang mit den Interessen des Benutzers. Eines der prominentesten Beispiele von unfreiwilliger Freigabe digitaler Identität ist die wachsende Anzahl an täglichem E-Mail Spam.

Ein essentieller Bestandteil zukünftiger Informationstechnologien ist daher das nutzergesteuerte Identitätsmanagement zum Schutz der Privatsphäre [CK01, CSS+06]. Das nutzergesteuerte Identitätsmanagementsystem muss verschiedene Anforderungen von mehreren Parteien erfüllen: es soll einerseits kontrollierte Pseudonymität und Anonymität des Nutzers gewährleistet sein, aber auch andererseits Verlässlichkeit der übermittelten Daten erreicht werden.

*Herr Pham wird durch die deutsche Exzellenzinitiative UMIC unterstützt.
†Herr Pimenidis wird vom 6. Rahmenprogramm der Europäischen Union unterstützt.

Um Pseudonymität und Anonymität zu erreichen muss die Verkettung von einem Pseudonym und seinem Inhaber geschützt sein. Darüber hinaus darf eine Verkettung von Pseudonymen des gleichen Inhabers, die in unterschiedlichen Rollen und/oder mit verschiedenen Kommunikationspartnern benutzt werden, im Regelfall nicht möglich sein. Um Zurechenbarkeit von Nutzern zu erreichen, muss es möglich sein Pseudonyme auf sichere Art zu authentisieren und basierend darauf zur Nutzung gewisser Dienste zu autorisieren. Im Falle einer beweisbaren Straftat müssen alle unter einem Pseudonym begangenen Handlungen zurechenbar und der Inhaber soweit erforderlich aufdeck- und haftbar sein.

Ein Credentialsystem mit solchen Eigenschaften wurde erstmals von David Chaum [Cha85] veröffentlicht. Weitere verbesserte Systeme wurden u.a. von Brands [Bra00] und Camenisch/Lysyanskaya [CL01] entwickelt. Die Grundidee ist wie folgt: Um Unverkettbarkeit zwischen mehreren Pseudonymen zu erreichen ist es notwendig, Attribute zwischen Pseudonymen transferieren zu können. Ein Nutzer sollte ein Attribut unter einem Pseudonym a von einer dritten Partei zertifiziert bekommen können und es dann unter einem anderen Pseudonym b vorzeigen können, ohne dabei einen Zusammenhang mit Pseudonym a herstellen zu müssen. Um dies zu erreichen, müssen die Attribute in Form von *anonymen Credentials* vorliegen. Das bedeutet, dass das Attribut durch eine dritte Instanz zertifiziert ist, das Zertifikat aber mit Hilfe kryptographischer Methoden unter verschiedenen Pseudonymen (für Empfänger und Dritte unverkettbar) vorgezeigt werden kann. Mehr Details dazu können in [Cha85, CE87, Che95, Dam90, LRSW00, CL01] nachgelesen werden.

Im Kontext von Identitätsmanagementsystemen werden Angriffe von Dritten oder Beteiligten ausgeführt, die über einen Nutzer weiterführende Informationen erhalten wollen [CSS$^+$06]. Das Identitätsmanagementsystem soll Individuen vor einer Analyse und Aufzeichnung ihrer Gewohnheiten und Daten schützen, selbst wenn ein Angreifer Großteile des Systems (z.B. alle Organisationen) überwachen und kontrollieren kann (siehe dazu Angreifermodell in [CL01]). Diese Anforderung ist bedingt erreichbar, da Angriffe existieren, die rein aus der Beobachtung der Aktionen eine Verkettung der Pseudonyme ermöglichen. In [PM06] wird solch ein Ansatz vorgestellt mit dem Namen „Consistent View Attack“ (CVA).

Der CVA basiert auf dem Konzept der kontextuellen Angriffe wie bei Raymond [Ray01] definiert und es existieren mehrere ähnliche Angriffe im Bereich der Anonymität (siehe z.B. Hitting Set Angriff [KAPR06]). Der Angriff beruht ausschließlich auf erlaubten Beobachtungen und Analysen dieser Beobachtungen.

CVA nutzt die logischen Zusammenhänge zwischen Zertifizierung und Vorzeigen der anonymen Credentials. Zum Beispiel ist es klar, dass ein anonymes Credential nicht gezeigt werden kann bevor es zertifiziert wurde. Die Nutzung dieser und weiterer logischen Korrelationen ist die Grundlage des Angriffs und der Basis für den Informationsgewinn des Angreifers. Konkret wird in [PM06] das Verkettbarkeitsproblem (linking problem) wie folgt definiert: Gegeben seien k Nutzer und die Beobachtung aller Zertifizierungs- und Vorzeigeaktionen. Kann der Angreifer korrekt alle Beobachtungen eindeutig in k disjunkten Untermengen zuordnen? Korrekt bedeutet hier, dass alle Aktionen in der Untermenge auch tatsächlich von demselben Nutzer getätigt wurden. Pashalidis und Meyer zeigen in ihrer Arbeit, dass das gestellte Problem NP-schwer ist, indem sie das Boolesche Erfüllbarkeitsproblem auf das gestellte Verkettbarkeitsproblem abbilden. Jedoch geben sie

keinen Algorithmus zur Lösung des Verkettbarkeitsproblems an. Dies ist Gegenstand dieser Arbeit.

Wir analysieren in dieser Arbeit Identitätsmanagementsysteme (insbesondere Idemix [CL01]) in ihrer stärksten Form, indem wir annehmen, dass alle Kommunikationen über eine anonyme Netzwerkebene geführt werden, und nur starke kryptographische Systeme eingesetzt werden. Desweiteren nehmen wir an, dass die Pseudonyme auf der Netztwerkebene unverkettbar sind. Somit werden alle Informationen aus Seitenkanäle ausgeblendet, die nicht über den Austausch von Pseudonymen und Credentials im System gewonnen werden. Es stellt sich heraus, dass selbst unter diesen Bedingungen ein beachtlicher Informationsverlust besteht. In manchen Fällen ist es sogar möglich die Pseudonyme eindeutig zu verketten. Da die meisten unserer Annahmen im Bezug auf die Praxis sehr stark sind, kann davon ausgegangen werden, dass der Informationsverlust in der Realität noch viel größer ist. Eventuell sind die Benutzer trotz der Benutzung eines Identitätsmanagementsystems schutzlos.

Vorerst werden wir diskrete mathematische Strukturen beschreiben, die grundsätzlich hinter Identitätsmanagementsystemen stecken und unser Modell vorstellen. Die praktische Evaluation wird in Zukunft folgen.

2 Theoretische Auswertung

In diesem Kapitel wird der Informationsverlust des Idemix-Systems modelliert und gemessen. Es wird gezeigt, dass Idemix informationstheoretisch unsicher ist, da es Fälle gibt, in denen die Pseudonyme der Benutzer eindeutig verkettet werden können. Dies bedeutet allerdings nicht zwangsläufig, dass das System auch in der Praxis unsicher sein muss. Dazu wird gezeigt werden, dass die eindeutige Verkettung der Pseudonyme im generellen Fall NP-schwer ist.

Unsere Arbeit unterscheidet sich von der Publikation [PM06] in zwei Hinsichten grundlegend. Zum einem ist das Modell ein wenig anders als beim CVA, so dass die Aussagen in [PM06] nicht in unserem Modell gültig sind. Zum anderen führen wir ein Verfahren zur Berechnung der logisch möglichen Verkettungen der Pseudonyme ein, welches in [PM06] fehlt. Der Vorteil dieses Verfahrens ist, dass es mit einer geringfügigen Veränderung auch zur Berechnung der Pseudonymverkettungen in [PM06] anwendbar ist.

Der in dieser Arbeit vorgestellte Angriff betrachtet lediglich einen passiven Angreifer. Dieser kann die Verkettung der Pseudonyme nur anhand von Zertifikatsausstellungs- und Vorzeigeaktionen schlussfolgern. Diese Ereignisse erscheinen, wenn ein Benutzer gemäß des Idemix-Protokolls Credentials von einer Organisation anfordert, oder sie dieser vorzeigt. Des Weiteren kann man auch aus den vorgeschriebenen Abläufen des Protokolls Regeln für die Zusammenhänge zwischen den Ereignissen erstellen, die die Anonymitätsmenge verringern. Diese Regeln werden im Kapitel 2.1 definiert.

Basierend auf diesen Regeln und den Beobachtungen der Ereignisse, wird im Kapitel 2.2 die formale Beschreibung des Lösungsraums des Idemix-Spiels in First-Order-Logic (FOL) vorgestellt. Es wird auch im Kapitel 2.2.3 ein Algorithmus eingeführt, der diese

Beschreibung berechnet.

Der Beweis, dass die eindeutige Zuordnung der Pseudonyme NP-schwer ist, wird im Anhang A gegeben.

2.1 Modellbeschreibung – Das Idemix-Spiel

Unser Modell des Idemix-Systems basiert auf den Beschreibungen in [CL01]. In diesem Spiel gibt es zwei Parteien. Diese bestehen aus der Menge aller k Benutzer, sowie die Menge aller Organisationen. Es wird angenommen, dass die Organisationen miteinander kooperieren, daher betrachten wir sie als eine Partei, nämlich die des *Angreifers*[1]. In diesem Modell führen die Benutzer eine Folge von Interaktionen mit dem Angreifer aus. Diese Aktionen werden von dem Angreifer protokolliert. Es wird angenommen, dass die Kommunikation über eine anonymisierende Netzwerkebene geführt wird, so dass der Angreifer keine weiteren Informationen zur Verfügung hat[2].

Das *Idemix Spiel* wird wie folgt modelliert: Der Angreifer kann insgesamt m verschiedene Typen von Credentials austeilen. Am Anfang des Spiels besitzen die Spieler keine Credentials. Diese können nur mittels *Issue*-Aktionen von dem Angreifer erworben werden, welcher diese Ereignisse aufzeichnet. Wenn ein Benutzer Credentials erhalten hat, dann kann er diese (oder eine Teilmenge davon) einer Organisation mittels einer *Show*-Aktion vorzeigen. Da alle Organisationen zusammenarbeiten und gemeinsam den Angreifer bilden, werden auch die Ereignisse zentral aufgezeichnet. Bei jedem Issue- oder Show- Ereignis muss der Benutzer eines seiner Pseudonyme vorzeigen. Wir bezeichnen die Menge der Pseudonyme jedes Benutzers i mit Q_i für $i \in \{1, \ldots, k\}$ und mit $P := \biguplus_{i:=1}^{k} Q_i$ die Menge aller Pseudonyme.

Die Ereignisse im Idemix-Spiel entsprechen der folgenden sechs *Grundregeln*:

Regel 1 Ein Benutzer kann nur dann ein Credential vorzeigen, wenn er es vorher erworben hat.

Regel 2 Credentials können nicht unter den Benutzern geteilt werden.

Regel 3 Unterschiedliche Benutzer können gleiche Credentials erwerben.

Regel 4 Die Benutzer bleiben die ganze Zeit auf der Netzwerkebene anonym.

Regel 5 Ein Benutzer kann dasselbe Credential mehrmals vorzeigen.

Regel 6 Ein Benutzer kann dasselbe Credential nicht mehrmals erwerben.

Die letzte Regel wird dadurch begründet, dass die Implementierung des Idemix- Systems es nicht vorsieht ein Credential an denselben Benutzer mehrmals zu vergeben. An dieser

[1] Die Annahme, dass alle Organisationen miteinander kooperieren stellt ein Worst Case Szenario für das Identitätsmanagementsystem dar, welches systembedingt existiert.

[2] Diese Annahme ist in der Realität nur schwer umzusetzen. Wir machen jedoch diese Annahme, um zu zeigen, dass selbst unter diesen erschwerten Umständen ein Angreifer Wissen gewinnen kann.

Stelle sei auch angemerkt, dass die Regelsätze des CVA's sich nur bezüglich der Regel 6 von unserem Idemix- Spiel unterscheidet. Dieser unscheinbare Unterschied hat allerdings einen signifikanten Einfluss auf die Struktur des Problems. Es ermöglicht zu einem durch die zusätzliche Information die Verbindung des Idemix-Spiels mit dem Knotenfärbungsproblem. Zu anderem bewirkt es, dass die Aussagen bezüglich der Komplexität des CVA's nicht auf das Idemix-Spiel übertragen werden können.

Das Ziel des Angreifers im Idemix-Spiel ist es, die Anonymitätsmenge anhand der Beobachtungen zu reduzieren. Wenn die Anonymitätsmenge so stark verkleinert werden kann, dass die Pseudonyme P eindeutig in k Segmente partitioniert werden können, dann sagen wir, dass der Angreifer das Spiel *gewonnen* hat; in diesem Fall hat er nämlich alle Pseudonyme eindeutig verkettet.

2.1.1 Beobachtungen des Angreifers

Die Beobachtungen des Angreifers haben auf Grund der zugrundeliegenden Protokolle in Idemix eine bestimme Struktur, deren Elemente wir jetzt genauer spezifizieren. Die Domäne der *Pseudonymmenge* P, der *Credentialmenge* C, sowie der *Ereignismenge* E sind wie folgt definiert:

$$P := \{p_1, \ldots, p_n\}, \text{ wobei } p_i \text{ Pseudonyme sind}$$
$$C := \{1, \ldots, m\}, \text{ wobei jede Nummer ein Credential darstellt}$$
$$E := \{\textit{issue}, \textit{show}\}, \text{ wobei } \textit{issue}, \textit{show} \text{ Ereignisse sind}$$

Eine *Beobachtung* O ist ein drei-Tupel $(P \times C \times E)$. Die Beobachtungen werden in einer chronologisch sortierte Liste zu einem *Protokoll* $\mathcal{H} := (O_1, \ldots, O_t)$ angelegt, die der Angreifer von Anfang an bis zum Zeitpunkt t sammelt.

Beispiel 2.1. *Sei $\mathcal{H}$ ein Protokoll, das den Grundregeln entspricht. Wir nehmen ferner an, dass der Angreifer folgende Beobachtungen notiert hat, die chronologisch von links nach rechts aufsteigend dargestellt sind:*

$$\mathcal{H} := ((p_1, 1, \textit{issue}), (p_2, 1, \textit{show}), (p_3, 2, \textit{issue}), (p_4, 2, \textit{show}), (p_5, 1, \textit{show}),$$
$$(p_5, 2, \textit{show}), (p_6, 3, \textit{issue}), (p_7, 3, \textit{issue}), (p_8, 1, \textit{show}), (p_8, 3, \textit{show}))$$

Die Menge an Pseudonymen besteht in diesem Beispiel aus acht Pseudonymen $P = \{p_1, \ldots, p_8\}$ und die Credentialmenge aus drei Elementen $C := \{1, 2, 3\}$.

Entsprechend des Protokolls zeigt die erste Beobachtung einen Benutzer mit dem Pseudonym p_1, der das Credential 1 erwirbt. Als nächstes zeigt ein Benutzer mit dem Pseudonym p_2, dass er das Credential 1 besitzt. Die letzten beiden Beobachtungen des Protokolls notieren einen Benutzer mit dem Pseudonym p_8, der die Credentials 1 und 3 vorweist.

Der Angreifer verfolgt das Ziel aus den Grundregeln und dem Protokoll Informationen bezüglich der Verkettung zwischen den Pseudonymen herzuleiten. Aus dem Beispiel 2.1 kann beispielsweise gefolgert werden, dass die Pseudonyme p_1 und p_2 zu demselben Benutzer gehören müssen. Den Grund dafür liefert Regel 1. Da p_1 das einzige Pseudonym

ist, das das Credential 1 erworben hat, bevor es durch p_2 gezeigt wird, müssen folglich die beiden Pseudonyme zu demselben Benutzer gehören.

2.2 Berechnung von Lösungen

Dieses Kapitel beschreibt eine Strategie, um aus den Grundregeln und einem gegebenen Protokoll einen First-Order-Logik (FOL) Term zu erzeugen, der die Lösungen des Idemix-Spiels beschreibt. Dabei ist eine *Lösung* $\mathcal{S} := \{Q_1, \ldots, Q_k\}$ eine aus den Grundregeln gefolgerte Partitionierung der Pseudonyme P, die aus den Show und Issue-Ereignissen gefolgert wurde. Ist eine Partitionierung hingegen widersprüchlich zu einer der Grundregeln oder dem gegebenen Protokoll, dann ist sie keine gültige Lösung und wird als *inkonsistent* bezeichnet. *Konsistente* (Lösungs-)Partitionierungen widersprechen hingegen keiner Beobachtung. Wir nennen die in einer Partition vorkommenden Mengen $Q_j \in \mathcal{S}$ *Segmente*. Alle Pseudonyme eines Segments werden demselben Benutzer zugeordnet.

2.2.1 Teilprotokolle

Unser Algorithmus zur Berechnung der Lösungen erfordert eine vorherige Bearbeitung des Protokolls. Dabei werden die Beobachtungen eines gegebenen Protokolls $\mathcal{H}$ bezüglich der Credentials in *Teilprotokolle* zusammengefasst. Jedes *Teilprotokoll* $\mathcal{H}_j$ ist ein chronologisch geordnetes Tupel von Beobachtungen, die alle Ereignisse mit Bezug auf das Credential j enthalten. Aus der Ordnung der Beobachtungen in jedem Teilprotokoll ergibt sich auch ihre relative Ordnung in dem Protokoll $\mathcal{H}$. Diese Unterteilung verändert nicht den Lösungsraum des Idemix-Spiels. In Bezug auf das Beispiel 2.1 sind die Credentials 1, 2 und 3; daher lauten die resultierenden Teilprotokolle:

$$\mathcal{H}_1 := ((p_1, 1, \textit{issue}), (p_2, 1, \textit{show}), (p_5, 1, \textit{show}), (p_8, 1, \textit{show}))$$
$$\mathcal{H}_2 := ((p_3, 2, \textit{issue}), (p_4, 2, \textit{show}), (p_5, 2, \textit{show}))$$
$$\mathcal{H}_3 := ((p_6, 3, \textit{issue}), (p_7, 3, \textit{issue}), (p_8, 3, \textit{show}))$$

Diese Unterteilung ist darin begründet, dass die Grundregeln nur Folgerungen zwischen Beobachtungen bezüglich desselben Credential erlauben. Zum Beispiel kann mit Regel 1 anhand der chronologischen Abfolge von Show- und Issue-Ereignissen zu demselben Credential eine Relation der beteiligten Pseudonyme hergeleitet werden (siehe Beispiel 2.1). Außerdem werden Relationen zwischen den Pseudonymen nicht verändert, da die chronologische Ordnung der Beobachtungen in jedem Teilprotokoll ihrer relativen Ordnung in dem Gesamtprotokoll entspricht.

2.2.2 Lösungen der Teilprotokolle

Eine Partitionierung der Pseudonyme P, die eine Lösung darstellt, muss konsistent zu dem Protokoll $\mathcal{H}$ sein. Folglich muss eine solche Lösung in unserem Beispiel auch konsistent zu den Teilprotokollen $\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3$ sein. Das heißt, dass man zu derselben Lösung

kommen kann, wenn man die Pseudonymverkettungen aus jeden Teilprotokollen (mit den Basisregeln) extrahiert und sie miteinander verknüpft.

Für jede Partitionierung gilt, dass die Pseudonyme p_i, p_j, die zu demselben Segment gehören, demselben Benutzer zugeordnet werden. Diese Pseudonyme sind somit in derselben *Äquivalenzklasse*[3] und wir schreiben daher $p_i \sim p_j$, wobei die Äquivalenzrelation $\sim$ symmetrisch, reflexiv und transitiv ist. Die Äquivalenzklasse, in der sich ein Pseudonym p_i befindet, wird mit $[p_i]$ referenziert. Zwei Äquivalenzklassen sind genau dann äquivalent $[p_i] = [p_j]$, wenn $p_i \sim p_j$ gilt. Mit den Grundregeln lassen sich folgende Informationen für jedes Teilprotokoll (in unserem Beispiel) herleiten:

$$\begin{array}{lll} \mathcal{H}_1: & (p_1 \sim p_2) \wedge (p_1 \sim p_5) \wedge (p_1 \sim p_8) & \Leftrightarrow ([p_1] = [p_2]) \wedge ([p_1] = [p_5]) \wedge ([p_1] = [p_8]) \\ \mathcal{H}_2: & (p_3 \sim p_4) \wedge (p_3 \sim p_5) & \Leftrightarrow ([p_3] = [p_4]) \wedge ([p_3] = [p_5]) \\ \mathcal{H}_3: & (p_6 \sim p_8 \vee p_7 \sim p_8) \wedge (p_6 \not\sim p_7) & \Leftrightarrow ([p_6] = [p_8] \vee [p_7] = [p_8]) \wedge ([p_6] \neq [p_7]) \end{array}$$

Man beachte, dass die Regel 6 zu keinem Informationsgewinn bei $\mathcal{H}_1$ und $\mathcal{H}_2$ führt, da in diesen Teilprotokollen nur jeweils ein Issue-Ereignis vorkommt. Anders verhält es sich bei $\mathcal{H}_3$. Hier werden zwei Pseudonyme p_6 und p_7 beobachtet, die das Credential 3 anfordern. Regel 6 impliziert hier, dass p_6 und p_7 nicht in einer Äquivalenzrelation zueinander stehen können, welches wir durch $(p_6 \not\sim p_7)$ ausdrücken.

Wenn nun die Informationen zu den Teilprotokollen miteinander verknüpft werden, kann des Weiteren gefolgert werden, dass die Pseudonyme in $\mathcal{H}_1$ und $\mathcal{H}_2$ zu derselben Äquivalenzklasse gehören. Dies ergibt sich aus der Transitivität von $\sim$. Nennen wir diese Äquivalenzklasse Q_1, wobei $p_1, p_2, p_3, p_4, p_5, p_8 \in Q_1$ gilt. Aus der Information von $\mathcal{H}_3$ kann entnommen werden, dass entweder die Äquivalenz $[p_6] = Q_1$ für den Fall von $p_6 \sim p_8$, oder die Äquivalenz $[p_7] = Q_1$ für den Fall $p_7 \sim p_8$ zutreffen muss. Hierbei handelt es sich um ein „exklusives oder“ wegen der Relation $(p_6 \not\sim p_7)$. Hierdurch bleiben nur noch folgende Lösungen übrig:

$$\mathcal{S}_1 := \{\{p_1, p_2, p_3, p_4, p_5, p_6, p_8\}, \{p_7\}\} \qquad \mathcal{S}_2 := \{\{p_1, p_2, p_3, p_4, p_5, p_7, p_8\}, \{p_6\}\}$$

In diesem Beispiel wurden die extrahierten Informationen auch zusätzlich als FOL Terme dargestellt. Dabei werden Verhältnisse von Äquivalenzklassen $[p_i]$ anstatt von Pseudonymen p_i beschrieben. Das ermöglicht es, die Lösungen des Idemix-Spiels über das Erfüllbarkeitsproblem zu ermitteln. Dazu betrachtet man die Äquivalenzklassen $[p_i]$ als Variablen, denen man einen Wert zuordnen kann, der einen eindeutigen Bezeichner eines Segments Q_j darstellt. Lösungen des Idemix-Spiels zu einem gegeben Protokoll $\mathcal{H}$ sind somit Variablenbelegungen, für die die FOL Terme von $\mathcal{H}_1, \mathcal{H}_2$ und $\mathcal{H}_3$ den Wert $true$ annehmen. Solche Variablenbelegungen *erfüllen* die FOL Terme, die von $\mathcal{H}$, bzw. von ihren Teilprotokollen beschrieben werden.

Als Beispiel betrachtet man die Lösung $\mathcal{S}_1$. Sei 1 der Bezeichner des ersten Segments dieser Lösung und 2 der Bezeichner des zweiten Segments. Belegt man alle Variablen, die den Äquivalenzklassen $[p_1], [p_2], [p_3], [p_4], [p_5], [p_6], [p_8]$ entsprechen mit dem Wert 1 und die Variable zu der Klasse $[p_7]$ mit dem Wert 2, dann sind alle FOL Terme der Teilprotokolle erfüllt. Die FOL Terme in diesem Beispiel repräsentieren den Lösungsraum zu dem

[3] Die Begriffe Segment und Äquivalenzklasse sind gleichbedeutend

gegeben Protokoll. Im Kapitel 2.2.3 stellen wir Algorithmen vor, die diese FOL Terme automatisch berechnen und daraus die Lösungen (wie im obigem Beispiel) bestimmen.

2.2.3 Berechnung der konjunktiven Normalform zu einem Protokoll

Im Kapitel 2.2.2 haben wir bereits ein Beispiel gesehen, wie die Beschreibung der Lösung zu einem Protokoll mit der Relation $\sim$ und den logischen Symbolen $\wedge$ und $\vee$ einfach in eine äquivalente *konjunktive Normalform* (KNF) umgewandelt werden kann. Wir werden hierzu einen Algorithmus vorstellen.

Zur Vereinfachung der Schreibweise werden nun bei den Äquivalenzklassen die Klammern [] um ein Pseudonym weggelassen, wenn die Bedeutung aus dem Kontext eindeutig ist. Die Darstellung p_i bezeichnet also jetzt die Variable zu der Äquivalenzklasse $[p_i]$, wenn nicht ausdrücklich auf etwas anderes hingewiesen wird. Der folgende Algorithmus berechnet die KNF zu einem Protokoll $\mathcal{H}$.

Der Algorithmus 1 wiederholt für jedes der m Credentials $i \in \{1, \ldots, m\}$ die folgenden drei Schritte:

1. Bestimmung des Teilprotokolls $\mathcal{H}_i$ zu dem Credential i.
2. Beschreibung der Äquivalenzrelationen zu $\mathcal{H}_i$ (mit Regeln 1, 2) durch die Klausel cl.
3. Bestimmung der Ungleichheitsrelationen zu $\mathcal{H}_i$ (mit Regel 6).

Die Berechnung des Teilprotokolls im ersten Schritt wird hier nicht angegeben, da es bereits im Kapitel 2.2.1 behandelt wurde. Der zweite Schritt wird in den Zeilen 4 bis 11 im Algorithmus 1 umgesetzt. Dabei wird die KNF cnf_i zu dem Teilprotokoll $\mathcal{H}_i$ mit Hilfe der Regeln 1 bis 5 bestimmt. Schritt drei wird in Zeile 12 ausgeführt und wendet die Regel 6 auf Pseudonyme an, die dasselbe Credential vorzeigen. Diese Pseudonyme werden untereinander mit Ungleichheitsrelationen versehen. Formal wird diese Information durch folgende Konjunktionen ausgedrückt:

$$\bigwedge_{(p_{i_u}, i, \textit{issue}), (p_{i_v}, i, \textit{issue}) \text{ in } \mathcal{H}_i, u \neq v} (p_{i_u} \neq p_{i_v}). \tag{1}$$

Ohne diesen Ausdruck (1) in Zeile 12, würde der Algorithmus 1 eine Beschreibung der Lösungen zu dem Consistent-View-Problem liefern.

Es ist zu beachten, dass der Term cnf zu dem Protokoll $\mathcal{H}$ nur Äquivalenz- und Ungleichungsrelationen zwischen Pseudonymen darstellt. Pseudonyme, die nicht in Relation zu anderen Pseudonymen stehen, sind nicht in den Term enthalten. Solche Pseudonyme nennen wir *kontextfrei*. Im Idemix Modell sind kontextfreie Pseudonyme jene, die in den Beobachtungen nur solche Credentials anfordern, die nicht von anderen Pseudonymen angefordert, oder gezeigt werden. Bei dem Consistent-View-Problem ist ein Pseudonym kontextfrei, wenn es ein Credential i anfordert, das nachher von keinem anderen Pseudonym vorgezeigt wird.

Um die kontextfreien Pseudonyme in der Berechnung der Lösungen zu berücksichtigen, speichert Zeile 14 die kontextgebundenen Variablen in der Menge P_{assign}. Das sind jene Variablen, die in cnf vorkommen. Diese Variablen werden *Kern-Pseudonyme* genannt. Die kontextfreien Variablen P_{free} entsprechen der Differenz zwischen den Variablen P in dem Protokoll und den Kern-Pseudonymen.

$$P_{free} := P \setminus P_{assign}, \tag{2}$$

Da die kontextfreien Pseudonyme P_{free} im Gegensatz zu den Kern-Pseudonymen P_{assign} keine nachweisbare Relation zu anderen Pseudonyme haben, können sie jeder beliebigen (bereits vorhandenen und noch nicht vorhandenen) Äquivalenzklasse zugeordnet werden. Daher bestehen die Lösungen zu dem Protokoll $\mathcal{H}$ aus den aus cnf hergeleiteten Partitionierungen kombiniert mit P_{free}. Wir nennen die Partition, die allein aus cnf ermittelt wird *Kernlösung* oder *Kernpartition*. Also ist eine Kernlösung ein Partitionierung der Menge P_{assign}, während eine Lösung eine Partitionierung der Menge P darstellen. Die Menge aller Kernpartitionen zu cnf wird der *Kern* genannt. Wenn $P_{free} = \emptyset$ dann enthält der Kern sogar alle Lösungen zu dem Protokoll $\mathcal{H}$.

Algorithm 1 HistoryToFOL

1: **procedure** HISTORYTOFOL($\mathcal{H}$)
2: $cnf := true$
3: **for** $i := 1$ to m **do**
4: $cnf_i := true$
5: **for** $j :=$ (first appear. of *show* in $\mathcal{H}_i$) to (last appear. in $\mathcal{H}_i$) **do**
6: $cl := false$
7: **for** each observation $(p_{i_l}, i, issue)$ in $\mathcal{H}_i$ with $((l < j) \wedge (i_l \neq i_j))$ **do**
8: $cl := cl \vee (p_{i_l} = p_{i_j})$
9: **end for**
10: $cnf_i := cnf_i \wedge cl$
11: **end for**
12: $cnf := cnf \wedge cnf_i \bigwedge_{(p_{i_u}, i, issue), (p_{i_v}, i, issue) \text{ in } \mathcal{H}_i, u \neq v} (p_{i_u} \neq p_{i_v})$
13: **end for**
14: $P_{assign} := \{p_i \mid p_i \text{ in } cnf\}$
15: **return** cnf, P_{assign}
16: **end procedure**

Das folgende Beispiel eines Protokoll $\mathcal{H}$, das aus zwei Teilprotokollen $\mathcal{H}_1$ und $\mathcal{H}_2$ besteht, illustriert den Fall, wenn $P_{free} \neq \emptyset$ gilt.

$$\mathcal{H}_1 := ((p_1, 1, issue), (p_2, 1, show)) \qquad \text{and} \qquad \mathcal{H}_2 := ((p_3, 2, issue)).$$

Der Algorithmus 1 liefert für diese Eingabe die Werte $cnf := (p_1 = p_2)$ und $P_{assign} := \{p_1, p_2\}$ zurück. Mit der Gleichung (2) erhält man $P_{free} := \{p_3\}$. Der Kern zu cnf besteht hier aus exakt einer Kernpartition $\mathcal{S} := \{\{p_1, p_2\}\}$. Die Lösungen, die sich aus der *Kombination* der Kernlösung mit P_{free} ergeben, lauten $\mathcal{S}_1 := \{\{p_1, p_2, p_3\}\}$ und $\mathcal{S}_2 := \{\{p_1, p_2\}, \{p_3\}\}$. Diese repräsentieren zugleich alle Lösungen zu $\mathcal{H}_1$.

Im Kapitel 2.2.2 wurde ein Beispiel gezeigt, wie der Kern zu dem KNF Term cnf bestimmt werden kann. Zur Überprüfung einer Hypothese, d.h. einer Partition, kann man cnf verwenden. Man generiert dabei eine Partition von P_{assign} und überprüft, ob die entsprechende Variablenbelegung cnf erfüllt. Eine Variablenbelegung, die cnf erfüllt, ist offensichtlich eine Kernlösung, während eine nicht erfüllende Belegung keine Kernlösung sein kann. Die Menge der möglichen Partitionen einer Menge lassen sich mit der Sterling-Formel zweiter Art wie in [GKP94] beschrieben, berechnen.

3 Experimentelle Resultate

In diesem Kapitel wenden wir die vorgestellten Algorithmen auf das Idemix Modell an, um den Informationsverlust im Bezug auf verschiedene Systemparameter zu studieren. Wir haben dafür eine Simulation geschrieben, die Beobachtungen generiert, die durch die Interaktionen der Benutzer mit dem Angreifer entstehen.

Für die Generierung der Beobachtung betrachten wir eine feste Menge von Benutzern mit einer variablen Anzahl an Pseudonymen und generieren dazu eine bestimmte Anzahl an Beobachtungen. Die Benutzer verfügen am Anfang über keine Credentials. Zu jeder Zeiteinheit wird ein Benutzer aus der Menge der Benutzer zufällig ausgewählt. Für diesen Benutzer bestimmen wir zunächst die relative Anzahl r an Credentials, die er erworben hat:

$$r = \frac{\text{Anzahl der Credentials des Nutzers}}{\text{Anzahl aller Credentialtypen}}.$$

Aus r ergibt sich die Wahrscheinlichkeit, dass ein Benutzer eines seiner vorhandenen Zertifikate vorzeigt, als

$$p_{show} = r^{\alpha},$$

wobei α ein Parameter ist, um unterschiedliche Benutzerverhalten zu modellieren. Entsprechend ist die Wahrscheinlichkeit, dass ein neues Zertifikat angefordert wird $p_{issue} = 1 - p_{show}$. Die Pseudonyme und Credentials, die der Benutzer bei einem Show- bzw. Issue-Ereignis auswählt, sind zufällig und gleichverteilt aus der Menge bereits erworbener bzw. noch nicht erworbener Credentials entnommen.

Mit dem Parameter α können wir das Benutzerverhalten variieren. Ein Wert $\alpha < 1$ simuliert einen Benutzer, der nur eine geringe Anzahl an Credentials in den ersten Runden ansammelt und sie dann kontinuierlich wiederholt vorzeigt. Hingegen stellt $\alpha > 1$ einen Benutzer dar, der möglichst viele Credentials ansammelt, bevor er sie vorzeigt.

Unsere Simulation geht davon aus, dass der Angreifer die Anzahl der Benutzer k kennt und nach einer eindeutigen Partitionierung der Pseudonyme in k Segmente sucht. Die entwickelten Algorithmen können auch angewendet werden, wenn k unbekannt ist, nur verlangt es aus simulativer Sicht viel mehr Laufzeit. Als Standardeinstellung verwendet die Simulation die Werte $u = 5$ Benutzer, $p = 3$ Pseudonyme pro Benutzer, $c = 10$ Credentialtypen, $o = 1000$ Beobachtungen und $\alpha = 1$. Um den Einfluss einzelner Parametern zu

verdeutlichen, variieren wir in den Simulationen jeweils nur eines der Parametern innerhalb eines bestimmten Intervalls. Die Ergebnisse sind in der Abbildung 1 zu finden. Alle Plots zeigen die Wahrscheinlichkeit eines erfolgreichen Angriffs, d.h. die Wahrscheinlichkeit, dass die Pseudonyme eindeutig verkettet werden können. Aus den ersten zwei Plots geht hervor, dass die Wahrscheinlichkeit eines erfolgreichen Angriffs mit der Anzahl der Benutzer sowie der Anzahl der Pseudonyme abnehmen. Hingegen steigt die Wahrscheinlichkeit eines Erfolgs mit der Anzahl der Credentials. Der vierte Plot zeigt den Einfluss des Benutzerverhaltens, das durch α modelliert wird. Wenn die Benutzer nur wenige Credentials gesammelt haben und diese sehr schnell vorzeigen, dann sind die Pseudonyme leichter verkettbar. Umgekehrt steigt die Privatsphäre der Benutzer, wenn sie so viele Credentials wie möglich sammeln und sie erst später zeigen.

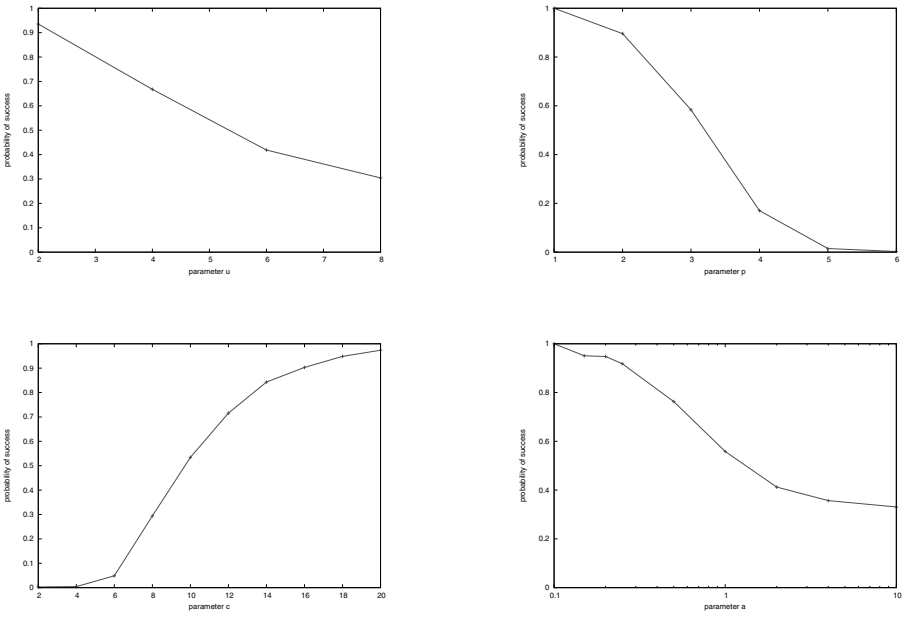

Abbildung 1: Experimentelle Resultate: Wahrscheinlichkeit eines erfolgreichen Angriffs.

4 Diskussion und Schlussfolgerung

In dieser Arbeit haben wir den Informationsverlust durch Issue-Show Ereignisse informations- und wahrscheinlichkeitstheoretisch untersucht. Dazu wurden Algorithmen eingeführt (Kapitel 2.2.3), um diese Informationen in FOL darzustellen. Unser Ansatz ist mit leichten Änderungen auch auf das Consistent-View Problem [PM06] anzuwenden. Aus

Platzmangel konnten wir dazu keine eigenen Analysen vorstellen. Durch die Reduktion des Idemix-Spiels auf das Knotenfärbungsproblem konnten wir auch zeigen, das dieser Angriff auf Idemix NP-vollständig ist (siehe Anhang A). Diese Reduktion zeigt ebenfalls, dass das Idemix Problem mit graphentheoretischen Ansätzen gelöst werden kann. Wir erwarten daher, dass über die Graphentheorie noch mehr Strukturen im Idemix-Spiel aufgedeckt werden können.

Die eingeführten Algorithmen wurden auch in unserer Simulation des Idemix- Systems eingesetzt. Die Resultate zeigen, dass in einigen Fällen die Pseudonyme eindeutig verkettet werden können, wenn die Anzahl der Benutzer bekannt ist. Die Kenntnis von k ist allerdings nicht immer notwendig, da auch ohne dieses Wissen Pseudonyme eindeutig verkettet werden können. In diesem Fall wäre lediglich die Wahrscheinlichkeit eines solchen Ereignisses geringer (und der Simulationsaufwand höher). Die Simulation zeigt zwar, dass es Situationen gibt, in denen Idemix kein Schutz bietet, allerdings kann der Benutzer durch sein Verhalten darauf Einfluss nehmen.

Informationstheoretische Betrachtungen wie durch Claude Shannon (siehe Unicity Distance Ansatz [Sha49] und [Hel77]) eingeführt und hier von uns eingesetzt, sind für die Analyse des maximal Möglichen relevant. Des Weiteren enthüllen sie diskrete Strukturen, die dem System immanent sind. Unser Ansatz hat gezeigt, dass das anonyme Credential im Prinzip, d.h. so wie es in Idemix eingesetzt ist, informationstheoretisch unsicher ist. Es bleibt allerdings noch offen, ob die generelle NP-Vollständigkeit des Idemix Problems eine lückenlose praktische Sicherheit darstellt. Können nicht triviale Instanzen des Idemix Problems existieren, die effizient lösbar sind? Das wollen wir in den zukünftigen Arbeiten behandeln.

Des Weiteren werden wir unser Modell durch die gewonnenen theoretischen Kenntnisse erweitern und es unter realistischeren Bedingungen evaluieren. Wir möchten z.B. herausfinden, wie sich das System verhält, wenn die Teilnehmer ein Credential x mal „issuen“ können, wobei $1 \leq x \leq n$ gilt und n eine obere Schranke darstellt. Ein weiteres Ziel ist es in Zukunft das Angreifermodell von einen passiven auf einen aktiven Angreifer zu erweitern.

Literatur

[Bra00] Stefan A. Brands. *Rethinking Public Key Infrastructures and Digital Certificates - Building in Privacy*. MIT Press, Cambridge-London, 2nd. Auflage, August 2000.

[Bro41] R. L. Brooks. On Colouring the Nodes of a Network. Jgg. 37 of *Cambridge Philos. Soc.*, Seiten 194 – 197, 1941.

[CE87] David Chaum und Jan-Hendrik Evertse. A Secure and Privacy-Protecting Protocol for Transmitting Personal Information Between Organizations. In *Proceedings on Advances in cryptology – CRYPTO '86*, Seiten 118–167, London, UK, 1987. Springer-Verlag.

[Cha85] David Chaum. Security without Identification: Transaction Systems to Make Big Brother Obsolete. *Commun. ACM*, 28(10):1030–1044, 1985.

[Che95] Lidong Chen. Access with Pseudonyms. In *Proceedings of the International Conference on Cryptography: Policy and Algorithms*, Seiten 232–243, London, UK, 1995. Springer-Verlag.

[CK01] Sebastian Clauß und Marit Köhntopp. Identity Management and its Support of Multilateral Security. *Compututer Networks*, 37(2):205–219, 2001.

[CL01] Jan Camenisch und Anna Lysyanskaya. An Efficient System for Non-transferable Anonymous Credentials with Optional Anonymity Revocation. In *Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT '01)*, Seiten 93 – 118, London, UK, 2001. Springer-Verlag.

[CSS+06] Sebastian Clauß, Stefan Schiffner, Sandra Steinbrecher, Dogan Kesdogan, Tobias Kölsch und Lexi Pimenidis. Identitätsmanagement und das Risiko der Re-Identifikation. In Jana Dittmann, Hrsg., *Sicherheit*, Jgg. 77 of *LNI*, Seiten 188–191. GI, 2006.

[Dam90] I. B. Damgard. Payment Systems and Credential Mechanisms with Provable Security Against Abuse by Individuals. In *CRYPTO '88: Proceedings on Advances in cryptology*, Seiten 328–335, New York, NY, USA, 1990. Springer-Verlag New York, Inc.

[DD006] Schwerpunktthema: Identität - eine sichere Sache? *Datenschutz und Datensicherheit - DuD*, 30(9):528 – 594, September 2006.

[GKP94] Ronald L. Graham, Donald E. Knuth und Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science (2nd Edition)*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1994.

[Hel77] M. E. Hellman. An Extension of the Shannon Theory Approach to Cryptography. *IEEE Trans. on Info. Theory*, IT-23:289–294, 1977.

[KAPR06] D. Kesdogan, D. Agrawal, V. Pham und D. Rauterbach. Fundamental Limits on the Anonymity Provided by the Mix Technique. *IEEE Symposium on Security and Privacy*, May 2006.

[LRSW00] Anna Lysyanskaya, Ronald L. Rivest, Amit Sahai und Stefan Wolf. Pseudonym Systems. In *SAC '99: Proceedings of the 6th Annual International Workshop on Selected Areas in Cryptography*, Seiten 184–199, London, UK, 2000. Springer-Verlag.

[PM06] Andreas Pashalidis und Bernd Meyer. Linking Anonymous Transactions: The Consistent View Attack. In George Danezis und Philippe Golle, Hrsg., *Proceedings of the Sixth Workshop on Privacy Enhancing Technologies (PET 2006)*, Cambridge, UK, June 2006. Springer.

[Ray01] Jean-François Raymond. Traffic analysis: protocols, attacks, design issues, and open problems. In *International workshop on Designing privacy enhancing technologies*, Seiten 10–29, New York, NY, USA, 2001. Springer-Verlag New York, Inc.

[Sha49] C. Shannon. Communication Theory of Secrecy Systems. *Bell Sys. Tech. J.*, 28:656–715, 1949.

A NP-Vollständigkeit des Idemix-Spiels

Mit der im Kapitel 2 eingeführten Strategie kann man die Lösungen des Idemix-Spiels berechnen. Unsere Simulationen haben gezeigt, dass es sogar Fälle gibt, in denen eine

eindeutige nicht-triviale Lösung gefunden werden kann. Es stellt sich daher die Frage, wie schwierig es ist, eine eindeutige Lösung zu einem Protokoll zu berechnen. Wir werden beweisen, dass die Berechnung der eindeutigen Lösung ein NP-vollständiges Problem darstellt.

Als Beweis präsentieren wir eine effiziente Reduktion des minimalen Knotenfärbungsproblems auf das Idemix-Spiel, die zeigt, dass das Idemix Problem mindestens so schwer wie das Färbungsproblem ist. Es ist bekannt, dass das minimale Knotenfärbungsproblem NP-vollständig ist [Bro41].

A.1 Reduktion der minimalen Knotenfärbung auf das Idemix-Spiel

Das minimale Knotenfärbungsproblem ist ein Graphenfärbungsproblem. Dabei gilt es die minimale Anzahl an Farben zu finden, um die Knoten eines ungerichteten Graphen so zu färben, dass adjazente Knoten unterschiedlichen Farben zugeordnet werden. Sei $G = (V, E)$ ein ungerichteter Graph mit der Knotenmenge $V = \{p_1, \ldots, p_n\}$ und der Kantenmenge $E \subseteq V \times V$. Ohne Beschränkung der Allgemeinheit nehmen wir an, dass es sich um einen zusammenhängenden Graphen handelt.

Zu jeder Kante $e_i \in E$ mit $e_i = \{p_{i_1}, p_{i_2}\}$ konstruiert der Reduktionsalgorithmus folgende Beobachtungen des Idemix-Spiels:

$$(p_{i_1}, issue, i), (p_{i_2}, issue, i) \quad \Longleftrightarrow \quad [p_{i_1}] \neq [p_{i_2}] \tag{3}$$

Sei $\mathcal{H}$ das Protokoll, das aus den Beobachtungen in (3) generiert wurde. Es ist offensichtlich, dass jede (Idemix) Lösung zu $\mathcal{H}$ eine Partitionierung von V darstellt. Durch die Konstruktion der Beobachtungen ist leicht zu erkennen, dass alle adjazenten Knoten p_{i_1} und p_{i_2} disjunkten Segmenten der Lösung zugeordnet werden.

Wenn jedes Segment einer Lösung mit einer anderen Farbe assoziiert wird, dann entspricht das genau einer Knotenfärbung von G. Insbesondere stellt die Lösung mit der geringsten Anzahl an Segmenten eine minimale Knotenfärbung dar. Aus dieser Reduktion geht hervor, dass das Idemix-Spiel mindestens NP-vollständig ist.

A.2 Verifizierbarkeit in polynomieller Zeit

Im Kapitel 2.2.2 wurde gezeigt, dass zu einem Protokoll effizient ein KNF-Term generiert werden kann. An dieser KNF kann dann eine Lösung verifiziert werden, indem überprüft wird, ob die KNF durch die Variablenbelegungen der Lösung erfüllt wird. Diese Überprüfung ist in polynomieller Zeit durchführbar. Probleme, deren Lösungen in polynomieller Zeit verifiziert werden können, gehören zu der Klasse der NP-Probleme, folglich muss das Idemix-Spiel NP-vollständig sein.