

Kundenkarten in hochdynamischen Systemen

Von einer Gefährdung zum Schutz der Privatsphäre

Martin Kähler und Rafael Accorsi

Abteilung Telematik
Institut für Informatik und Gesellschaft
Albert-Ludwigs-Universität, Freiburg
{kaehler, accorsi}@iig.uni-freiburg.de

Zusammenfassung. Mit Kundenkarten erhalten Kunden Vergünstigungen als Anreiz für wiederholte Geschäftsbeziehungen. Dabei hinterlassen sie jedoch unbemerkt Profile, die Unternehmen zur Überwachung des Kaufverhaltens verwenden, um z.B. vor dem nächsten Einkauf gezielt Marketing zu betreiben [1]. Um auf die Kunden schon *während* des Einkaufs einzuwirken, versucht der stationäre Handel nun die bereits in e-Commerce erfolgreich eingesetzten Techniken zur individuellen Kundenbetreuung – etwa Recommender Systeme und personalisierte Dienste – zu übertragen. Das Ubiquitous Computing bietet hierfür die notwendigen Technologien und verwandelt Einkaufsläden in hochdynamische Informationssysteme, in denen jedoch Kunden aufgrund der Kontexterfassung noch umfangreichere Profile ermöglichen, die ihnen mittels der Kundenkarten leicht zuzuordnen sind. Um dieser Bedrohung der Privatsphäre entgegenzutreten, wird ein konzeptioneller Ansatz vorgeschlagen, mit dem Kundenkarten auch zum potenziellen Schutz der Privatsphäre in hochdynamischen Systemen verwendet werden können. Schwerpunkt dieses Beitrags ist die Erweiterung der Policy-Sprache NAPS um Operatoren zum Vergleich zweier Richtlinien mittels Differenz, die eine notwendige Grundlage für die Realisierung dieses Vorhabens bildet.

1 Der Laden und die Privatsphäre der Kunden

Ubiquitous Computing steht für eine Vielfalt an Technologien, die im Grunde Mobilität der Teilnehmer (Mobile Computing), Wahrnehmung bzw. Einbeziehung von Kontext (Pervasive Computing) und Autonomie interner Vorgänge (Autonomous Computing) liefern [2]. Diese Kernbereiche werden zukünftig weiter zusammenwachsen und gemeinsam eine neue Form von Informationssystemen ermöglichen, die so genannten *hochdynamischen Systeme* (HDS) [3].

Mit dem Ziel, in ihren stationäre Läden eine eins-zu-eins-Beziehung mit den Kunden aufzubauen, ist der Handel einer der Branchen, die die bevorstehenden Vorteile eines HDS-Einsatzes erkannt haben und die dafür erforderlichen Techniken weiterentwickeln [4]. Die ersten Versuche dahingehend sind bereits im Gange: In Deutschland entwickelt die METRO-Gruppe den “Future Store”. Hier werden Einkaufswagen mit Personal Shopping Assistants (PSA) versehen, d.h. Rechnern

mit Barcode-Leser und WLAN-Verbindung als Schnittstelle zum Informationssystem des Ladens. Über ein PSA können Dienste – etwa Recommender-Systeme, Werbung und Rabatte – angeboten und mittels der Kundenkarten personalisiert werden. Andere Konzernen setzen zusätzlich auf Kontextdaten von RFID-Tags, Kameras und Sensoren, um Bewegung oder gar Mimik der Kunden auszuwerten.

So vielversprechend sich die Zukunft der HDS-Läden anhört, so zeigen sich bereits vor ihrer tatsächlichen Realisierung Probleme hinsichtlich der Bedrohungen der Privatsphäre, die die Kundenakzeptanz dieser Systeme untergraben könnten. Denn die umfassende Erhebung verschiedenster Kontextdaten ist zwar einerseits essenziell für das Angebot von Diensten, führt andererseits jedoch zu erheblichen Eingriffen in der Privatsphäre der Kunden [4]. Somit werden Kunden in HDS-Läden noch größerer Überwachung ausgesetzt als bisher, wobei gegenwärtig keine einsetzbare Technologie zum Schutz ihrer Privatsphäre vorhanden ist.

2 Kundenkarten zum Schutz der Privatsphäre

Um auch in einer solch dynamischen Umgebung seine Privatsphäre schützen zu können, müssen Kunden die Möglichkeit erhalten, für ihre Daten den Grad der Erhebung und Art der Verwendung selbst zu bestimmen. Oft können Kunden jedoch ihr eigenes Verhalten nicht antizipieren: Wie Umfragen gezeigt haben, verfolgen viele bzgl. ihrer Privatsphäre einen sehr pragmatischen Ansatz und passen ihre Sicherheitsvorstellungen in Abwägung persönlicher Vorteile auch spontan der aktuellen Situation bzw. den aktuellen Kommunikationspartnern an [5]. Hier bietet sich die mitgeführte Kundenkarte als Lösung an. Speichert sie neben personenbezogenen und persönlichen Daten insbesondere individuelle Privacy-Richtlinien, kann der Kunde die Sammlung und weitere Verwendung seiner Daten aktiv regeln. Mithilfe seines PSA vor Ort kann deren Regeln seinen individuellen Anforderungen anpassen und so auf aktuelle Dienstangebote reagieren.

2.1 Privacy-Richtlinien für Kundenkarten

Eine Sprache für die Privacy-Richtlinien der Kundenkarten unterliegt gewissen Anforderungen an ihre Ausdrucksstärke. Kontrolle über die Datenherausgabe reicht nicht mehr, Datenkonsumenten müssen zusätzlich auch Verpflichtungen bzw. Obligationen aufgebürdet werden können, um auch die anschließende Verwendung aktiv zu regeln und z.B. die Löschung der gesammelten Daten nach Beendigung des Einkaufs zu verlangen [6]. Für die Formulierung solcher Regeln wurden bereits eine Reihe Sprachen entwickelt, z.B. das von dem W3C für den Einsatz im Web entwickelte P3P [7] oder IBM's EPAL [8] für den Enterprise-internen Einsatz. Die Herausforderung liegt vielmehr bei ihrer Verwendung in offenen HDS, in dem eine Vielzahl unterschiedlicher Richtlinien aufeinander treffen und so besondere Operatoren zur Analyse und Manipulation erfordern.

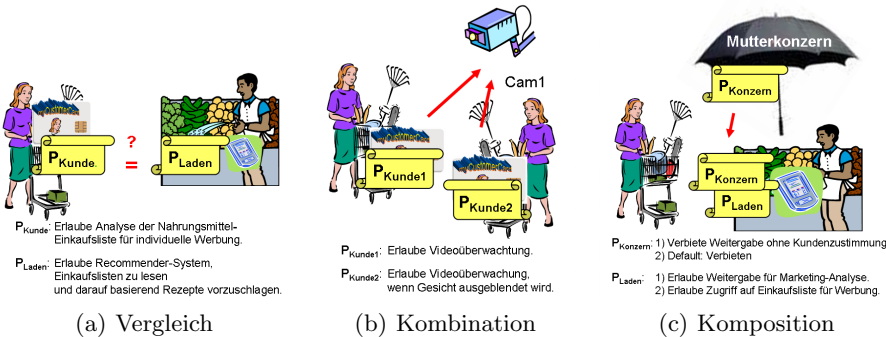


Abb. 1. Szenarien für Operationen auf Richtlinien

Ein Kunde muss Funktionen zur Verfügung haben, mit denen er Richtlinien auf ihre Aussagen hin überprüfen kann, z.B. ob die Richtlinien des Ladens seinen Ansprüchen in Form der eigenen Richtlinien genügen, oder worin sie sich unterscheiden. Lauten die Richtlinien wie in Abb. 1(a), sollte der Kunde feststellen können, dass der Verkäufer nach dessen Richtlinie nicht nur auf die Einkaufsliste für Nahrungsmittel, sondern auf alle Listen zugreifen darf, beispielsweise auch die für die Apotheke um die Ecke. Mit dem Wissen um diese Differenz kann der Kunde nun entscheiden, ob er auf den Dienst des Verkäufers verzichten will, oder doch die komplette Einkaufsliste preiszugeben.

Weitere, für HDS notwendige Operationen sind Verknüpfungen, die eine modulare Zusammensetzung aus Teilrichtlinien erlauben. Befinden sich z.B. zwei Kunden mit jeweils individuellen Richtlinien wie in Abb. 1(b) zur gleichen Zeit im Erfassungsraum einer Videokamera, muss diese die Richtlinien beider Kunden gleichwertig berücksichtigen, hier also Videoaufzeichnung mit anschließender Anonymisierung. Ebenso sollte eine Verknüpfung unterschiedlich priorer Richtlinien möglich sein, mit der Kunden z.B. ihre allgemeinen Richtlinien spezifisch für den Laden um die Ecke verfeinern oder auf Händlerseite wie in Abb. 1(c) ein Mutterkonzern unternehmensweite Mindestanforderungen vorgeben kann.

2.2 Formalisierung von Richtlinien mittels NAPS

Für den Einsatz im HDS-Laden eignet sich das von Raub und Steinwand entwickelte NAPS (Novel Algebraic Privacy Specification) Rahmenwerk [9], das auf EPAL bzw. auf dessen algebraischen Fundament E-P3P basiert.

Eine NAPS-Richtlinie besteht aus einer Liste priorisierter Regeln. Eine Regel besteht aus einem den Scope der Regel begrenzenden Wächter aus (*Benutzer* $U \times$ *Aktionen* $A \times$ *Daten* $D \times$ *Zugriffszweck* P), einer Menge von Bedingungen und der Entscheidung, die diese Regel schließlich liefert. Die Elemente eines Wächters werden zur intuitiven Spezifizierung und Aggregation in hierarchischen Strukturen (H, \leq) partiell geordnet, wie in der Benutzerhierarchie (*Kassierer* \leq *Verkauf*

\leq *Laden* \geq *Marketing*). Für den Einbezug von Kontext α greift NAPS auf eine dreiwertige Prädikatenlogik C zurück und erlaubt so die für HDS vorteilhafte Auswertung auch unvollständiger Information.

NAPS betrachtet Obligationen nicht als reine Blackbox-Anweisungen, sondern besitzt ein gegenüber E-P3P erweitertes Obligationsmodell $(O, \leq, \wedge, \top, \perp)$ aus einem Halbverband mit maximalem Element \top , der leeren Obligation, und minimalem Element \perp , der unerfüllbaren Obligation. Auferlegen der Obligation \top bedeutet, dass eine Aktion ohne weitere Verpflichtung ausgeführt werden kann; auferlegen von \perp , dass eine Aktion nicht ausgeführt werden darf. Für den Einsatz auf der Kundenkarte wählen wir einen Halbverband $(\mathfrak{Pot}(\tilde{O}), \subseteq)$ über der Potenzmenge der elementaren Obligationen \tilde{O} , der Mengenvereinigung \cup als Konjunktion, $\top := \emptyset$, $\perp := \tilde{O}$ und für $A, B \in \mathfrak{Pot}(\tilde{O}) : A \leq B \Leftrightarrow A \supseteq B$.

Die Entscheidung einer NAPS-Regel wird durch ein Paar von Obligationen $r = (o^+, o^-) \in O \times O$ beschrieben, wobei o^+ die Obligationen bei Ausführung der Aktion vorgibt, und o^- die Obligationen bei Nichtausführung. Die Auswertung einer Richtlinie muss somit nicht zwingend vorschreiben, ob eine Aktion erlaubt ist oder nicht. Die endgültige Entscheidung kann so dem Anfragenden überlassen, und der für das Zusammenspiel von HDS-Diensten notwendige Entscheidungsraum offen gehalten werden. So bedeutet beispielsweise die Entscheidung $(\{\text{benarichtigeKunden}\}, \{\text{halteDaten}\})$ für den Kassierer auf Anfrage, den RFID-Tag des verkauften Artikels zu zerstören: erlaubt, wenn der Kunde informiert wird. Alternativ kann er den Tag nicht zerstören, muss dann jedoch die Artikeldaten für eventuelle Reklamationen im Backend-Server weiter vorhalten. Mit diesen Definitionen sieht eine NAPS-Richtlinie \mathcal{P} wie folgt aus:

- (1, $p \leq$ (Kamera, aufnehmen, Kunde, Marktanalyse), hatZustimmung(Kunde), ($\{\text{anonymisiere}\}, \perp$))
- (2, $p \leq$ (Kassierer, kill, RFID, Verkauf), istElektro(Artikel), ($\{\text{benarchtigeKunden}\}, \{\text{halteDaten}\}$))
- (default $\{\perp, \top\}$)

Die Semantik der Richtlinien Sprache wird durch die Auswertungsfunktion $eval_\alpha(\mathcal{P}, q)$ für eine Anfrage $q \in (U \times A \times D \times P)$ bezüglich Richtlinie \mathcal{P} und aktuellen Belegung α der Kontextparameter bestimmt. Diese Funktion wertet die Regeln entsprechend ihren Prioritäten aus: ist der Regelwächter im Anfrage-Scope, wird die Regelentscheidung als final zurückgegeben. Ist ein Wächter nur aufgrund fehlender Kontextinformation nicht im Scope, wird die Entscheidung dieser Regel konjunktiv mit späteren Entscheidungen verknüpft, und die Auswertung unter Vermerk des nicht finalen Status fortgesetzt. Im Falle einer Unsicherheit aufgrund einer nur partiellen Parameterbelegung wird so grundsätzlich eine restriktiver Entscheidung, niemals jedoch eine freizügigere gefällt. Trifft keine Regel zu, wird eine Default-Entscheidung zurückgegeben.

Das aufwendige Mitschleifen des Auswertungsstatus ermöglicht NAPS die Abgeschlossenheit seiner Verknüpfungsoptoren Konjunktion und Komposition. Die Konjunktion $\mathcal{P}_1 \wedge \mathcal{P}_2$ wertet dabei \mathcal{P}_1 und \mathcal{P}_2 mit gleicher Priorität aus, während die Komposition $\mathcal{P}_1 \parallel \mathcal{P}_2$ bei der Auswertung \mathcal{P}_1 eine höhere Priorität verleiht, sie also zuerst auswertet.

2.3 Vergleich und Differenz in NAPS

Für den Vergleich bzw. Äquivalenz zweier Richtlinien definiert NAPS folgendes: Zwei Richtlinien sind äquivalent, wenn sie für alle möglichen Anfragen und Belegungen der Umgebungsparameter die gleichen Entscheidungen liefern. Ähnlich definiert NAPS die Verfeinerung, die überprüft, ob eine Richtlinie restriktiver bzw. spezifischer als eine andere ist, diese also in allen Punkten erfüllt.

Für Umgebungen wie den HDS-Laden sind diese beiden Tests jedoch nicht ausreichend, denn gerade hier wird der Benutzer mit vielen Diensten und ihren jeweiligen Einzelrichtlinien konfrontiert, und es ist nicht anzunehmen, dass alle die gleichen oder höchstens restriktiver als die Kundenentscheidungen fällen werden. Wie soll sich der Kunde verhalten, wenn die Richtlinie eines Dienstes nicht seiner eigenen entspricht, sie also wie z.B. in Abb. 1(a) nicht äquivalent oder restriktiver ist? Ebenso muss er Richtlinien vergleichen, wenn er Veränderungen an ihnen über die Zeit nachvollziehen will. Jetzt muss der Kunde eingreifen und herausfinden, welche Rechte in welchen Situationen von ihm zusätzlich verlangt werden, damit der Dienst zur Ausführung kommen kann.

Bisher ist diese Analyse nur mühselig von Hand zu bewerkstelligen. Der bereits für E-P3P vorgeschlagene Differenzoperator [10] erlaubt es dem Kunden, die Richtlinien auf genau die Teile zu reduzieren, die er für seine weiteren Entscheidungen benötigt, nämlich all die Teile, die zusätzliche Berechtigungen ausgeben bzw. schwächere Verpflichtungen aussprechen. Hier wird nun der Differenzoperator auf NAPS übertragen:

Definition 1. *Seien zwei Richtlinien \mathcal{P}_1 und \mathcal{P}_2 mit gleichem Vokabular gegeben. Dann ist die Differenz \mathcal{P}_{2-1} die Regelmenge, die für alle Anfragen q und Belegungen α der Umgebungsparameter die gleichen Entscheidungen liefert wie \mathcal{P}_2 , reduziert die Regeln, für deren Entscheidung mit $(o_i^+, o_i^-) = \text{eval}_\alpha(\mathcal{P}_i, q)$ eine der folgenden Eigenschaften gilt: $o_2^+ \not\subset o_1^+$ oder $o_2^- \not\subset o_1^-$.*

Eine erste Implementierung kann über einen Brute-Force-Ansatz geschehen. Es müssen für beide Richtlinien \mathcal{P}_1 und \mathcal{P}_2 die Entscheidungen für alle möglichen Anfragen und alle möglichen Belegungen der Umgebungsparameter berechnet werden. Eine Regel mit dem Scope der Anfrage (also entsprechendem Wächter) und den Bedingungen und Entscheidung der jeweils zutreffenden Regel von \mathcal{P}_2 wird in \mathcal{P}_{2-1} aufgenommen, wenn die Mengen o_2^+ oder o_2^- andere oder weniger Verpflichtungen aufweisen als die entsprechende Regel von \mathcal{P}_1 .

Dieser Brute-Force-Ansatz testet alle möglichen Kombinationen von Anfragen und Parameterbelegungen, so dass sein Aufwand exponentiell mit dem verwendeten Vokabular wächst. Für den praktischen Gebrauch ist daher ein effizienterer Algorithmus wünschenswert, wie ihn Backes et al. für den effizienten Test auf Richtlinienverfeinerung in [11] vorgeschlagen. Die Ausnutzung der Semantik von NAPS, insbesondere der hierarchischen Strukturen, zur Reduzierung des Berechnungsaufwandes der Differenz bleibt ebenso Teil zukünftiger Arbeit wie die Berücksichtigung von Vokabularerweiterungen.

3 Fazit

Dieser Artikel schlägt Kundenkarten als Technologie zum Schutz der Privatsphäre von Kunden vor. Mit auf ihnen gespeicherten Privacy-Richtlinien können Kunden Erhebung und Verwendung personenbezogener Kontextinformationen aktiv mitbestimmen. Wesentliche Voraussetzung hierfür ist eine Richtlinienprache entsprechender Ausdrucksstärke. Hier wird die Sprache NAPS um Operatoren erweitert, die der automatischen Visualisierung der Differenz zwischen Richtlinien dienen. Dieser Ansatz ist Teil eines Projekts zur Erhaltung der Privatsphäre in hochdynamischen Systemen. Um Kunden die Visualisierung der Richtliniendifferenzen verständlich zu machen, muss diese Information entsprechend aufbereitet werden, z.B. als Zeiger, der im Gefährdungsfall auch die tatsächlichen Unterschiede aufzeigt [12]. Für die Bereitstellung so genannter “privacy evidence” mit Information über erhobene Daten und deren Verwendung [4] sind solche Kundenkarten essenziell, können Richtlinien doch sicheren Logging-Verfahren [13] aufzuzeichnenden Ereignisse vorgeben. Gegenwärtig wird an einer Simulation gearbeitet, in der dieses Zusammenspiel praktisch untersucht werden soll.

Literaturverzeichnis

1. K. Albrecht. Supermarket cards: The tip of the surveillance iceberg. *Denver University Law Review*, 79(4):534–539, 558–565, 2002.
2. R. Accorsi, S. Höhn, M. Kähler, G. Müller, and M. Strasser. Sicherheit im Ubiquitous Computing: Schutz durch Gebote. In “Der Computer im 21. Jahrhundert: Perspektiven, Technologien, Auswirkungen”. Springer-Verlag, 2007. To appear.
3. G. Müller. Privacy and security in highly dynamic systems. *Communications of the ACM*, 49(9):28–31, 2006.
4. S. Sackmann, J. Strüker, and R. Accorsi. Personalization in privacy-aware highly dynamic systems. *Communications of the ACM*, 49(9):32–38, 2006.
5. L. Cranor, P. Guduru, and M. Arjula. User interfaces for privacy agents. *ACM Transactions on Computer-Human Interaction*, 13(2):135 – 178, 2006.
6. J. Park und R. Sandhu. The UCON_{ABC} usage control model. *ACM Transactions on Information and System Security*, 7(1):128–174, 2004.
7. W3C. Platform for privacy preferences – P3P. <http://www.w3.org/P3P/>.
8. P. Ashley, S. Hada, G. Karjoth, C. Powers, und M. Schunter. Enterprise privacy authorization language. Technical report, IBM Research, 2003.
9. D. Raub und R. Steinwandt. An algebra for enterprise privacy policies closed under composition and conjunction. In G. Müller (Hrsg.), *Proc. of the ETRICS*, vol. 3995 of LNCS, pp. 130–144. Springer-Verlag, 2006.
10. M. Kähler. Difference in security policies for dynamic systems. In R. Accorsi et al. (Hrsg.), *Proc. of SecAS Workshop*, vol. 183 of CEUR. RWTH-Aachen, 2006.
11. M. Backes, G. Karjoth, W. Bagga, und M. Schunter. Efficient comparison of enterprise privacy policies. In *Proc. of SAS*, pp. 375–382. ACM Press, 2004.
12. S. Höhn. Bringing users back into control: A new paradigm for usability in highly dynamic systems. In S. Fischer-Hübner et al. (Hrsg.), *Proc. of TrustBus*, vol. 4083 of LNCS, pp. 114–122. Springer-Verlag, 2006.
13. R. Accorsi. On the relationship of privacy and secure remote logging in dynamic systems. In S. Fischer-Hübner et al. (Hrsg.), *Proc. of the IFIP Security and Privacy in Dynamic Environments*, vol. 201 of IFIP, pp. 329–339. Springer-Verlag, 2006.