# Verteilte Autorisation in RFID-Ereignissystemen

Eberhard Grummt[1,2] · Martin Schöffel[1,2]

[1]SAP Research CEC Dresden
{eberhard.oliver.grummt | martin.schoeffel}@sap.com

[2]Technische Universität Dresden, Professur Rechnernetze

## Zusammenfassung

Durch die Integration RFID-basiert erfasster Ereignisdaten aus mehreren verteilten Repositorys können im Kontext des Supply Chain Managements neuartige Anwendungen realisiert werden. Der Zugriff auf die einzelnen Repositorys muss jedoch durch Zugriffskontrollsysteme beschränkt werden. Da die Weitergabe lokal erfasster Ereignisdaten auch potenziell die Sicherheitsanforderungen von Handelspartnern tangiert, müssen entsprechende Zugriffsentscheidungen verteilt getroffen werden können. Ausgehend von einer Problemanalyse beschreibt dieser Beitrag eine auf XACML basierende Rahmenarchitektur namens aidXACML sowie die domänenspezifische Ressourcenadressierungssprache EAL.

# 1 Einführung

Globale „Traceability Networks“ sind verteilte Systeme, mit denen sich der aktuelle und alle vorherigen Aufenthaltsorte und Zustände von physikalischen Handelsobjekten bestimmen lassen [ACKS06]. Sie basieren auf der automatischen Erkennung dieser Objekte mittels Auto-ID-Technologien wie z. B. RFID an bestimmten Punkten in einer Lieferkette, üblicherweise mindestens dem Warenein- und ausgang jeder beteiligten Firma. Die Sichtungen von Objekten lösen *Ereignisse* (engl. *Events*) aus, die in Datenbanken gespeichert werden, welche auch als *Auto-ID-Repositorys* bezeichnet werden. Es wird im Folgenden von einer lokalen Datenhaltung ausgegangen, d. h. jede beteiligte Firma betreibt ein eigenes Auto-ID-Repository, in welchem ausschließlich selbst erfasste Ereignisse gespeichert werden. Entsprechend müssen zur Rückverfolgung eines Objekts Daten aus mehreren Quellen integriert werden. Die lückenlose Rückverfolgbarkeit steht allerdings im Zielkonflikt mit dem Wunsch der Firmen nach Geheimhaltung bestimmter Informationen (vgl. Autorisierungsautonomie [dVS96]). Deshalb muss jede beteiligte Firma bei jedem Zugriff auf ihr Auto-ID-Repository eine Zugriffskontrolle durchführen, um nur jene Informationen preiszugeben, für die der Anfragende ausreichende Berechtigungen besitzt. Die Bestimmung solcher Berechtigungen sowie ihre Durchsetzung bergen etliche Herausforderungen, welche in Abschnitt 2 vorgestellt und diskutiert werden. Grundlagen und verwandte Arbeite werden in Abschnitt 3 vorgestellt. Abschnitt 4 führt ein auf XACML basierendes Architektur-Rahmenwerk ein, das verteilte Zugriffsentscheidungen auf großen Mengen von Auto-ID-Ereignissen ermöglicht. In Abschnitt 5 wird ein Ansatz zur Spezifikation von entsprechenden Berechtigungen und Delegationen diskutiert, bevor in Abschnitt 6 eine Zusammenfassung und ein Ausblick gegeben werden.

P. Horster (Hrsg.) · D•A•CH Security 2008 · syssec (2008) 337-345.

# 2 Zugriffskontrolle in Traceability Networks

Die grundlegenden, im vorigen Abschnitt vorgestellten Sachverhalte sind in Abb. 1 illustriert. Ein Handelsobjekt $o$ hat während seines Weges durch die Lieferkette jeweils den Warenein- und ausgang von zwei Firmen sowie den Wareineingang einer dritten Firma passiert. Durch Integration der Ereignisse $Ereignis_1$ bis $Ereignis_5$, welche zu den Zeiten $t_1$ bis $t_5$ von den Firmen $Firma_1$ bis $Firma_3$ an RFID-Lesepunkten erfasst wurden, kann durch einen Anfragenden potenziell die Historie des Objekts $o$ ($Trace_o$) ermittelt werden. Abhängig von den Zugriffs-

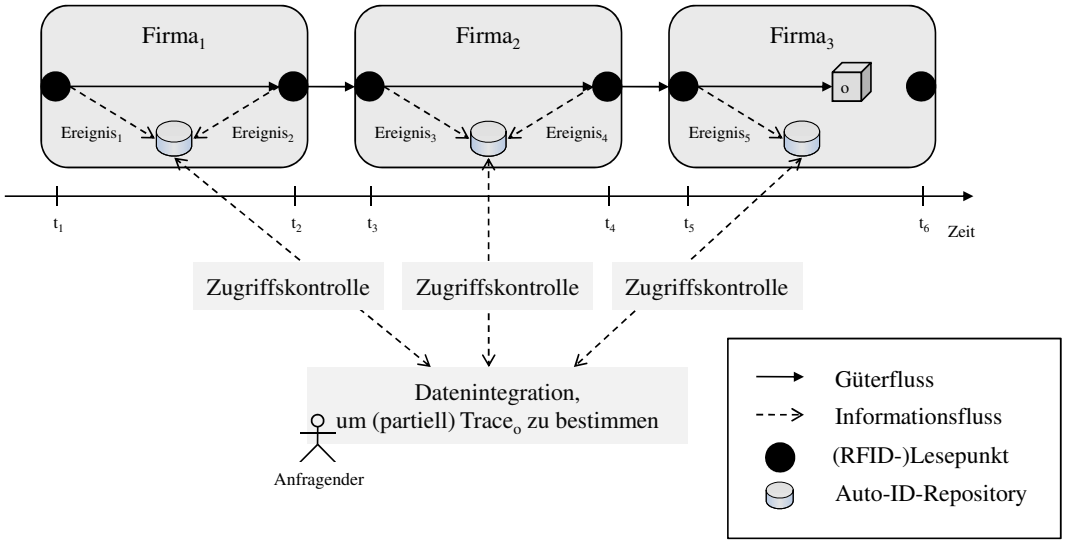


**Abb. 1:** Prinzip eines Traceability Networks

berechtigungen, die ihm in den einzelnen Repositorys gewährt werden, ist $Trace_o$ jedoch unvollständig. Schließlich ermöglichen die in Auto-ID-Repositorys gespeicherten Informationen umfangreiche Rückschlüsse z. B. über Produktions- oder Lagerkapazitäten, Lieferbeziehungen und letztendlich auch über den Umsatz von Unternehmen. Entsprechend sind Firmen daran interessiert, die Informationsweitergabe auf ein Maß zu beschränken, bei dem die wirtschaftlichen Vorteile die Nachteile des Vertraulichkeitsverlusts überwiegen. Die Untersuchung, wie dieses Maß ermittelt werden kann, ist nicht Gegenstand des vorliegenden Beitrags. Stattdessen werden technische Ansätze diskutiert, die Firmen sowohl autonom als auch gemeinschaftlich in die Lage versetzen können, die gewünschte Weitergabe von Ereignissen zu spezifizieren und durchzusetzen. Gegenüber klassischen Zugriffskontrollsystemen ergeben sich im Kontext der Traceability Networks folgende spezifische Herausforderungen bzw. Anforderungen:

- Die durch das Zugriffskontrollsystem zu schützenden Ressourcen sind *Ereignisse*, welche aus einzelnen Attributen bestehen und Zustandsänderungen an bestimmten Zeitpunkten repräsentieren. Durch ihre große, stetig wachsende Menge wäre ein Zugriffskontrollsystem, das auf diskreter Adressierung einzelner Ereignisse basiert (z. B. in Form von *Access Control Lists* [Lam71]), nicht handhabbar. Stattdessen müssen *Teilmengen* der erfassten Ereignisse flexibel, z. B. über logische Regeln, spezifiziert werden können.
- Firmen wünschen eine *feingranulare* Datenweitergabe, d. h. die Offenlegung muss nicht nur für komplette Ereignisse, sondern auch für deren einzelne Attribute steuerbar sein.
- Von einer Firma erfasste Ereignisse können Informationen enthalten, deren Preisgabe die

Vertraulichkeitsanforderungen anderer Firmen verletzen würden (z. B. durch Aufdeckung einer Lieferbeziehung durch Dritte). Deshalb können Zugriffsentscheidungen nicht ausschließlich lokal getroffen werden, sondern müssen potenziell die Autorisierungs-Anforderungen von Handelspartnern mit einbeziehen. Technisch bedeutet dies, dass die Durchsetzung von Berechtigungen von ihrer Entscheidung abgekoppelt werden muss, auch über Unternehmensgrenzen hinweg. Weiterhin muss auch die Kombination von potenziell sich widersprechenden oder erweiternden Autorisationen unterstützt werden.

- Anfragen an ein Auto-ID-Repository werden derart gestellt, dass durch die Angabe von einschränkenden Kriterien die gewünschten Ereignis-Teilmengen spezifiziert werden. Das Zugriffskontrollsystem hat die Aufgabe sicherzustellen, dass unberechtigt angefragte Ereignisse und Attribute nicht in der Ergebnismenge enthalten sind. Die einfachere

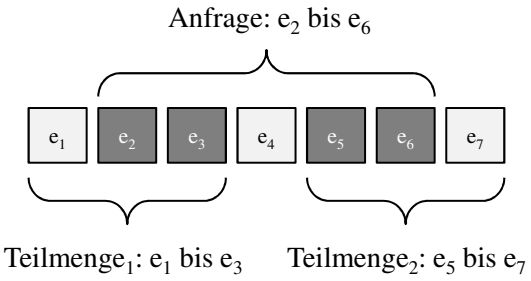


**Abb. 2:** Eine Anfrage, zwei Teilmengen einer Berechtigung und die auszuliefernden Ereignisse

Alternative, nämlich die Ablehnung von zu weit gefassten Anfragen, ist aus Gründen der Benutzbarkeit nicht akzeptabel. Schließlich müssten Anfragende dann genau ihre Berechtigungen kennen und ggf. viele einzelne Anfragen stellen um diese nicht zu verletzen. Ebenfalls nicht praktikabel ist die Manipulation der Anfrageparameter durch das Zugriffskontrollsystem (z. B. durch Einschränkung der zulässigen Parameter einer Web Service-Schnittstelle). Die Abfrageschnittstelle eines Repositorys könnte beispielsweise die Parameter `EventIDFrom` und `EventIDTo` zur Selektion von Ereignis-Teilmengen anhand fortlaufender ID-Bereiche anbieten. Da die Berechtigungen sich aber nicht notwendigerweise durch fortlaufende Bereiche beschreiben lassen, kann die einfache einschränkende Veränderung der Anfrageparameter nicht verwendet werden. Dies ist in Abb. 2 dargestellt: die auszuliefernde Ereignis-Teilmenge (dunkelgrau) kann nicht durch einen fortlaufenden Bereich von IDs (und damit nicht durch Werte für `EventIDFrom` und `EventIDTo`) ausgedrückt werden.

- Verbreitete Zugriffskontrollsysteme unterstützen für einzelne Anfragen nur deren Ablehnung oder deren Genehmigung. Das gilt insbesondere auch für Systeme, die eine Entkopplung von Durchsetzung und Entscheidung von Berechtigungen unterstützen (vgl. XACML [Mos05]). Würde ein solches System für die feingranulare Zugriffskontrolle in Auto-ID-Repositorys eingesetzt werden, müsste an die Entscheidungskomponente für jedes Attribut jedes einzelnen angefragten Ereignisse eine Anfrage gesendet werden. Dies ist aus Gründen der Performanz nicht akzeptabel. Entsprechend muss die zu entwerfende Entscheidungskomponente die *teilweise* Genehmigung von Anfragen unterstützen. Um eine effiziente Selektion der dem Anfragenden auszuliefernden Ereignisse aus der Gesamtmenge der Events zu ermöglichen, sind Mechanismen zur Übersetzung der Autorisationsantwort in Anfragen an relationale und XML-Datenbanken vorzusehen.

- Aufgrund der Dynamik zukünftiger Lieferketten, insbesondere der großen Anzahl potenzieller Handelspartner sowie der Unvorhersehbarkeit des konkreten Lieferwegs von Objekten sind Anfragende einem Auto-ID-Repository evtl. nicht in Form von lokalen Benutzerkonten und assoziierten Rechten bekannt. Deshalb müssen verteilte Authentifizierung und Autorisierung unterstützt werden. Da von Auto-ID-Repositorys auch Abonnements bestimmter Ereignisse durch externe Partner unterstützt werden sollen, ist das zeitversetzte Genehmigen von Anfragen, z. B. nach manuellem Bestätigen eines Administrators, vorzusehen. Dies kann außerdem helfen, den Administrationsaufwand zu reduzieren, da der Anfragende Berechtigungen vorschlagen kann, deren Genehmigung (ggf. reduziert) für den Systembetreiber weniger Arbeit darstellt als die Berechtigungen selbst zu definieren.
- Da noch nicht abzusehen ist, welche Authentifizierungs- und Autorisationsmethoden den Anforderungen in dynamischen Lieferketten am besten gerecht werden (z. B. *Possession Centric Access Management* [GA08]), ist die flexible Einbindung und Erweiterung entsprechender neuer Mechanismen vorzusehen. Daraus resultiert auch die Notwendigkeit, dem Anfragenden Rückmeldungen über die Art der unterstützten Mechanismen zu liefern. Ebenfalls denkbar ist die Rückmeldung von Informationen darüber, wie eine zunächst abgelehnte Anfrage durch Beibringung zusätzlicher Sicherheitsinformationen (*Credentials*) durch den Client doch noch genehmigt werden kann.

# 3 Grundlagen und verwandte Arbeiten

Das *EPCglobal Architecture Framework* [ABB+07] ist eine Sammlung von Technologie-Standards, mit deren Hilfe sich Traceability Networks bereits heute teilweise realisieren lassen. EPC steht für *Electronic Product Code*, die entsprechende Spezifikation umfasst mehrere Schemata für die weltweit eindeutige Nummerierung physischer Objekte [EPC06]. Die *EPC Information Services* (EPCIS) [EPC07] stellen eine Spezifikation für den Austausch von Auto-ID-basiert erfassten Ereignissen dar. Sie adressiert Ereignistypen sowie Erfassungs- und Abfrageschnittstellen. Aspekte der Sicherheit im Allgemeinen und der Zugriffskontrolle im Speziellen werden in den EPCglobal-Spezifikationen nur am Rande behandelt. Dies ist insofern verständlich, dass die inneren Arbeitsweisen konkreter Systeme nicht von den Standards vorgeschrieben werden, sondern lediglich Schnittstellen, Austauschformate und Protokolle definiert werden. Wie ein Zugriffskontrollsystem für EPCIS-konforme Systeme (sog. *EPCIS Repositorys*) realisiert werden kann ist eine wichtige, aber offene Fragestellung.

XACML (*eXtensible Access Control Markup Language*) [Mos05] ist ein Framework, das eine XML-basierte Sprache für Zugriffsrechte, eine Referenzarchitektur sowie ein ebenfalls XML-basiertes Request-/Response-Format für Zugriffsrechtsanfragen und -antworten umfasst. Die Zugriffsentscheidungen werden auf einer domänenunabhängigen Ebene getroffen, lediglich der *Policy Enforcement Point* (PEP) ist eine domänenspezische Komponente, die diese Entscheidungen durchsetzt. Die Entscheidungen selbst werden durch Auswertung von *Rules*, den elementaren Berechtigungs-Regeln, von einer vom PEP entkoppelten Komponente, dem *Policy Decision Point* (PDP) getroffen. Der PDP erhält vom PEP ein *XACML-Request* und generiert anhand von XACML-Berechtigungsdefinitionen ein entsprechendes *XACML-Response*. Welche Berechtigungsdefinitionen dabei überhaupt anwendbar sind, wird durch sogenannte *Matching-Algorithmen* ermittelt. In der XACML-Berechtigungsdefinitionssprache lassen sich Elemente der Typen `PolicySet`, `Policy` und `Rule` zur Definition von Berechtigungen

schachteln. Über `Target`-Elemente wird in diesen Elementen durch Angabe von Werten für `Subject`, `Resource`, `Action` und `Environment` definiert, auf welche Kombinationen von Subjekten, Objekten, Aktionen und Umgebungsbedingungen sie sich beziehen. Eine anwendbare `Rule` liefert genau ein Ergebnis (`Permit` oder `Deny`). In `Policy`- und `PolicySet`-Elementen werden sogenannte *Combining-Algorithmen* referenziert, welche die Ergebnisse mehrerer enthaltener und anwendbarer `Rule`- bzw. `Policy`-Elemente zusammenfassen. Durch Anwendung der Combining-Algorithmen werden auch für anwendbare `Policy`- und `PolicySet` sowie letztendlich für gesamte XACML-Requests Ergebnisse ermittelt, die jeweils entweder `Permit`, `Deny`, `Indeterminate` oder `NotApplicable` lauten.

GeoXACML [Mat04] bettet die Sprache GML (Geography Markup Language) in XACML ein, um domänenspezifische Anfragen und Policys definieren zu können. Des Weiteren werden Matching- und Combining-Algorithmen für GML definiert. Verteilte Autorisation hingegen wird nicht berücksichtigt, ebensowenig das teilweise oder verspätete Genehmigen von Anfragen. [FPP+02] beschreibt ein Konzept zur Rollen-Delegation in verteilten Systemen. Das vorgestellte Konzept „Distributed Role-based Access Control“ ist auch für ein Traceability Network denkbar, in dem potenziell unbekannte Teilnehmer Ereignisdaten abfragen. Abendroth und Jensen [AJ03] diskutieren das Konzept des teilweisen „Outsourcings“ von Sicherheitsmechanismen. Die effiziente Durchsetzung feingranularer Rechte auf RDBMS-basierten EPCIS Repositorys wird in [GM08] untersucht.

# 4 Architektur für verteilte Zugriffsentscheidungen

Im Folgenden wird eine verteilte Systemarchitektur vorgestellt, welche die Basis für die Erfüllung der in Abschnitt 2 aufgeführten Anforderungen bildet. Die Trennung der Komponenten zur Entscheidung über Zugriffsversuche (Policy Decision Point - PDP) und der Durchsetzung dieser Entscheidungen (Policy Enforcement Point - PEP) wird bereits in [YPG00] erwähnt und stellt auch ein grundlegendes Konzept der Systemarchitektur von XACML dar. Allerdings unterstützen Entscheidungs-Antworten gemäß der XACML-Spezifikation weder das teilweise noch das zeitversetzte Genehmigen von Anfragen. Auch das Delegieren von Entscheidungskompetenz wird nicht explizit unterstützt. XACML bietet als OASIS-Standard jedoch zahlreiche Vorteile, so dass die hier unter dem Namen *aidXACML* (*Auto-ID XACML*) vorgestellten Konzepte eine Erweiterung des XACML-Frameworks darstellen.

Abb. 3 gibt eine Übersicht über die wesentlichen konzeptionellen Bestandteile *Auto-ID-Repository* (z. B. EPCIS-basierend), *Lokaler Policy Decision Point* und *Entfernter Policy Decision Point*. Das Auto-ID-Repository implementiert aus XACML-Sicht den Policy Enforcement Point (PEP). Repository und PDP werden über Web Service-Schnittstellen angesprochen. Das Repository kann beispielsweise das *EPCIS Query Control Interface* [EPC07, S. 59] implementieren, über das Ereignisse abgerufen werden können.

Im Folgenden wird der Ablauf der Anfrageverarbeitung beschrieben. Der Referenzmonitor des Repositorys formt Zugriffsanfragen von Clients (1) zunächst in das aidXACML-Request-Format um und sendet diese als Parameter einer Web Service-Operation an den PDP (2). Dieser prüft zunächst mit Hilfe des EAL-Matching-Algorithmus', welche lokalen aidXACML-Policys auf die jeweilige Anfrage anwendbar sind. Diese werden dann ausgewertet und je nach vorgegebenem Combining-Algorithmus kombiniert. Liegen lokale Definitionen zur Delegation der Entscheidungsgewalt über bestimmte Ereignismengen vor, werden entsprechende aidXACML-Requests an die einzubeziehenden entfernten PDPs gesendet (3). Anschließend werden die

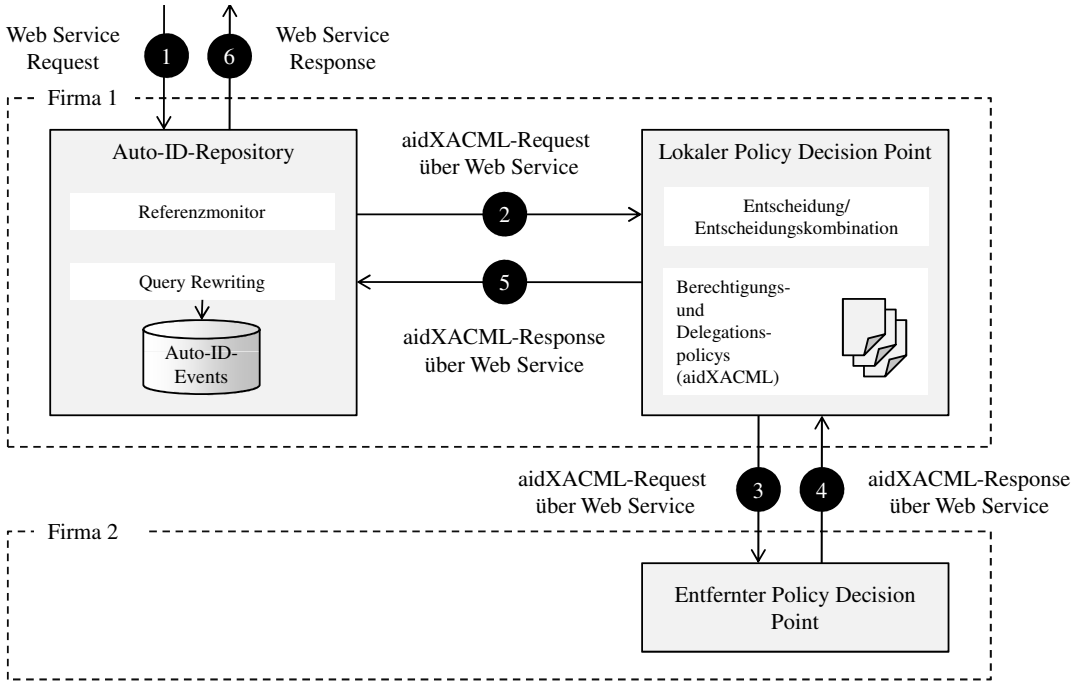


**Abb. 3:** Architekturübersicht und Ablauf einer Anfrage

Ergebnisse (4) je nach vorgegebenem Combining-Algorithmus mit den lokalen Entscheidungen kombiniert. Der PDP integriert die resultierende, in EAL vorliegende Entscheidung in ein aidXACML-Response und liefert diese als Antwort zurück (6). Im Falle der teilweisen oder gesamten Genehmigung der ursprünglichen Client-Anfrage wandelt das Repository die EAL-Regeln der Response in eine Anfrage an die Datenbank um (z. B. SQL oder XQuery), welche die Auto-ID-Ereignisse enthält. Das eigentliche „Enforcement“, d. h. die Durchsetzung der Zugriffsentscheidung des PDP, wird also von der dem Repository zugrundeliegenden Persistenztechnologie durchgeführt. Dies stellt sicher, dass trotz der Anforderungen der Attribut-Granularität und der Trennung von PEP und PDP auch die Anforderung der hohen Performanz erfüllt werden kann.

# 5 Regelbasierte Ereignisadressierung

Ein wesentlicher Bestandteil von aidXACML ist die neu entwickelte *Event Addressing Language* (*EAL*), mit der sich Ereignismengen regelbasiert spezifizieren lassen. EAL wird verwendet, um Ressourcen in aidXACML-Policys, -Delegationsregeln, -Requests und -Responses zu spezifizieren. EAL-Instanzen können in XML dargestellt werden und werden in dieser Form hauptsächlich in das XACML-`Resource`-Element eingebettet.

In EAL werden Ereignismengen (`EventSet`) durch die Angabe von zu vereinigenden Teilmengen spezifiziert. Diese Teilmengen (`EventSubset`) werden durch die Spezifikation von einer Basismenge, konjunktiv zu verknüpfenden Bedingungsmengen (`ConditionSets`) sowie einer Liste von sichtbaren Attributen definiert. Ein `ConditionSet` bezieht sich auf genau ein Attribut. Unter Verwendung der Elemente `Condition` und `Predicate` können für dieses Attribut einschränkende Bedingungen in disjunktiver Normalform formuliert werden. Dies lässt sich folgendermaßen formulieren:

$EventSet := EventSubset_1 \cup \ldots \cup EventSubset_l$
$EventSubset(VisibleAttrs, BasicSet) := ConditionSet_1 \wedge \ldots \wedge ConditionSet_m$
$ConditionSet(Attribute) := Condition_1 \vee \ldots \vee Condition_n$
$Condition := Predicate_1 \wedge \ldots \wedge Predicate_o$
$Predicate := Operator\ Operands$

Abb. 4 gibt ein Beispiel für die XML-Repräsentation einer EAL-Instanz. Sie drückt aus, dass eine Ereignis-Teilmenge aus der Grundmenge `ObjectEvent` selektiert wird, in der alle Attribute (*) sichtbar sein sollen und bei denen das Attribut `readPoint` den Wert `urn:epc:id:gid:100.200.300` hat. Die Einschränkung, dass sich ein `ConditionSet` nur auf genau ein Attribut beziehen kann wurde getroffen, damit sich zwei EAL-Instanzen so logisch verknüpfen lassen, dass Redundanzen entfernt werden können. Lautet beispielsweise ein Prädikat $a > 3$ und ein anderes, konjunktiv verknüpftes Prädikat $a > 6$, so soll nur $a > 6$ im Ergebnis enthalten sein. Würden sich Bedingungen allerdings aus komplexen prädikatenlogischen Ausdrücken zusammensetzen, so ließen sich derartige Optimierungen nicht effizient berechnen.

```
<eal:EventSet>
  <eal:EventSubset VisibleAttributes="*"
    basicSet="ObjectEvent">
    <eal:ConditionSet Attribute="readPoint"
      DataType="http://www.w3.org/2001/XMLSchema#string">
      <eal:Condition>
        <eal:Predicate Operator="equal">
          urn:epc:id:gid:100.200.300
        </eal:Predicate>
      </eal:Condition>
    </eal:ConditionSet>
  </eal:EventSubset>
</eal:EventSet>
```

**Abb. 4:** Beispiel für ein EAL-Dokument

Für die Bestimmung, ob ein aidXACML-`PolicySet`/-`Policy`/-`Rule`-Element auf einen konkreten `Request` anwendbar ist, wird der neue Matching-Algorithmus `eal-overlaps` definiert. Die Definition von XACML-`Response`-Nachrichten wird erweitert, so dass nicht mehr nur das komplette Genehmigen oder Ablehnen eines `Requests` unterstützt werden, sondern auch das teilweise Genehmigen. Dazu wird ein `Resource`-Element in `Response`-Nachrichten aufgenommen und ein neuer Wert für `Decision` namens `PartialPermit` eingeführt. Dies hat auch Auswirkungen auf die Combining-Algorithmen, die als Eingabe nun nicht nur die vier XACML-Decision-Typen erhalten können, sondern auch EAL-Instanzen. Aus diesem Grund werden Operatoren definiert, mit denen Disjunktionen (`eal-union`), Konjunktionen (`eal-intersect`) usw. von zwei EAL-Instanzen berechnet werden können. Diese Operatoren bilden auch die Basis für die Ermittlung von EAL-Instanzen, die Zugriffs- und Delegationsentscheidungen repräsentieren. Liegen die (kombinierten) Zugriffsregeln als $EAL_1$

und eine Anfrage als $EAL_2$ vor, so berechnen sich die Beschreibung für die auszuliefernden Ereignisse als $EAL_3$ = `eal-intersect`($EAL_1$,$EAL_2$)

Über EAL-Instanzen kann in aidXACML-Policys weiterhin eine Delegation der Zugriffsentscheidung für die jeweils spezifizierte Ereignismengen realisiert werden. Dazu werden die URL des externen Policy Decision Points sowie weitere Informationen wie das Protokoll-Binding angegeben. Da dieser externe Policy Decision Point seinerseits wiederum eingehende Anfragen weiterleiten kann, kann er auch als eine Art Auffindungsdienst für anwendbare Policies genutzt werden. Zur Laufzeit wird über Tickets sichergestellt, dass durch Fehlkonfiguration entstandende Delegationszyklen erkannt werden und nicht zu Endlosschleifen führen.

# 6 Zusammenfassung und zukünftige Arbeit

In diesem Beitrag wurden Anforderungen an Zugriffskontrollsysteme in globalen Traceability Networks im Allgemeinen und in Auto-ID-Repositorys im Speziellen sowie entsprechende Umsetzungsansätze diskutiert. Als wesentliche Anforderungen wurden die regelbasierte Adressierung von Ereignismengen, die attributgenaue Steuerung der Ereignis-Weitergabe, die automatische Einbeziehung von Handelspartnern in Zugriffskontrollentscheidungen sowie die automatische Reduktion von Anfragen identifiziert. XACML wurde als Basis für ein neuartiges Zugriffskontroll-Rahmenwerk namens aidXACML ausgewählt. In aidXACML werden vor allem Schwachstellen von XACML in den Bereichen der Ressourcenadressierung, dem teilweisen Genehmigen von Anfragen sowie dem Einbeziehen externer Zugriffsentscheidungen adressiert. Da in aidXACML Zugriffsentscheidungsprozesse nicht nur Genehmigung, Ablehnung und verschiedene Fehlertypen zurückliefern können, mussten auch neue Combining-Algorithmen entwickelt werden. Einen tragende Rolle in der internen Verarbeitung von Anfragen, Antworten sowie Zugriffs- und Delegationsregeln spielt die neu entwickelte Event Addressing Language EAL. Mit EAL-Instanzen können Ereignismengen regelbasiert feingranular adressiert werden. Mit Hilfe von EAL-Operatoren können Zugriffs- und Delegationsentscheidungen getroffen werden, sie stellen auch die Basis für neue Matching- und Combining-Algorithmen in aidXACML dar. EAL kann in SQL und XQuery übersetzt werden, so dass eine effiziente Durchsetzung entsprechender Zugriffsregeln realisiert werden kann.

Die weiterführende Arbeit wird sich mit der Frage beschäftigen, wie Geschäftskontext-Informationen wie z. B. Geschäftstransaktionen verwendet werden können, um aidXACML-Zugriffspolicys auf Basis übergeordneter Regeln automatisch zu generieren.

# Literatur

[ABB+07] Felice Armenio, Henri Barthel, Leo Burstein, Paul Dietrich, John Duker, John Garrett, Bernie Hogan, Oleg Ryaboy, Sanjay Sarma, Johannes Schmidt, KK Suen, Ken Traub, and John Williams. The EPCglobal Architecture Framework – EPCglobal Final Version 1.2. http://www.epcglobalinc.org/standards/architecture/architecture_1_2-framework-20070910.pdf, September 2007.

[ACKS06] Rakesh Agrawal, Alvin Cheung, Karin Kailing, and Stefan Schönauer. Towards Traceability across Sovereign, Distributed RFID Databases. In *IDEAS '06: Proceedings of the 10th International Database Engineering and Applications Symposium*, pages 174–184, Washington, DC, USA, 2006. IEEE Computer Society.

[AJ03] Joerg Abendroth and Christian D. Jensen. Partial Outsourcing: A New Paradigm for Access Control. In *SACMAT '03: Proceedings of the eighth ACM symposium on Access control models and technologies*, pages 134–141, New York, NY, USA, 2003. ACM Press.

[dVS96] Sabrina De Capitani di Vimercati and Pierangela Samarati. Access Control in Federated Systems. In *NSPW '96: Proceedings of the 1996 workshop on New security paradigms*, pages 87–99, New York, NY, USA, 1996. ACM Press.

[EPC06] EPCglobal Inc. EPCglobal Tag Data Standards Version 1.3 – Ratified Specification, March 2006.

[EPC07] EPCglobal Inc. EPC Information Services (EPCIS) Version 1.0 Specification. http://www.epcglobalinc.org/standards/EPCglobal_EPCIS_Ratified_Standard_12April_2007_V1.0.pdf, April 2007.

[FPP+02] Eric Freudenthal, Tracy Pesin, Lawrence Port, Edward Keenan, and Vijay Karamcheti. dRBAC: Distributed Role-based Access Control for Dynamic Coalition Environments. In *ICDCS '02: Proceedings of the 22 nd International Conference on Distributed Computing Systems (ICDCS'02)*, page 411, Washington, DC, USA, 2002. IEEE Computer Society.

[GA08] Eberhard Grummt and Ralf Ackermann. Proof of Possession: Using RFID for large-scale Authorization Management. In Max Mühlhäuser, Alois Ferscha, and Erwin Aitenbichler, editors, *Constructing Ambient Intelligence: AmI-07 Workshops Proceedings*, LNCS, pages 174–182. Springer-Verlag Berlin Heidelberg, 2008.

[GM08] Eberhard Grummt and Markus Müller. Fine-grained Access Control for EPC Information Services. In Christian Floerkemeier and Marc Langheinrich, editors, *The Internet of Things 2008*, volume 4952 of *LNCS*, pages 35–49. Springer-Verlag Berlin Heidelberg, March 2008.

[Lam71] Butler Lampson. Protection. In *Proceedings of the 5th Annual Princeton Conference on Information Sciences and Systems*, pages 437–443, Princeton University, 1971.

[Mat04] Andreas Matheus. *Declaration and Enforcement of Access Restrictions for Distributed Geospatial Information Objects*. PhD thesis, Technische Universität München, 2004.

[Mos05] Tim Moses. eXtensible Access Control Markup Language (XACML) Version 2.0. http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf, February 2005.

[YPG00] R. Yavatkar, D. Pendarakis, and R. Guerin. A Framework for Policy-based Admission Control (RFC 2753). http://www.ietf.org/rfc/rfc2753.txt, January 2000.