Rahmenarchitektur und Sicherheitsinfrastruktur der deutschen Gesundheitstelematik-Plattform – die MDA Methodologie

Bernd Blobel
Universitätsklinikum Magdeburg

Verzahnte Versorgung erfordert eine adäquate Unterstützung durch Informationssysteme, die offen, skalierbar, flexibel, portabel, intelligent interoperabel, vertrauenswürdig und zukunftsfähig sein müssen. Die modellgetriebene Architektur der Object Management Group mit ihrer Komponentenorientierung, der Trennung der logischen von den technologischen Aspekten und einer Metasprachen-Umgebung, die Sicherung gemeinsamer Vokabularien und die Einbeziehung der unterschiedlichen Domänenexperten mit ihren Ausdrucksmitteln bieten die Voraussetzungen für eine Rahmenarchitektur bzw. entsprechende Lösungsarchitekturen für die Gesundheitstelematik-Plattform einschließlich ihrer Sicherheitsinfrastruktur, die die beschriebenen Anforderungen erfüllen. Der Beitrag beschreibt die Architekturprinzipien zukunftsfähiger Gesundheitsinformationssysteme, geht auf einige Details ein und analysiert grob, wieweit die gegenwärtigen Projekte zur Schaffung einer Gesundheitstelematik-Plattform in der Bundesrepublik diesen Anforderungen genügen.

Einleitung

Zukunftssichere Gesundheitsinformationssysteme und Gesundheitsnetze werden zunehmend auf der Basis von Komponentensystemen entwickelt, wobei die Spezifikation der Komponenten sowie ihre Aggregation durch generische Modelle beschrieben werden. Dazu werden in internationalen Projekten und Gremien Prototypen, Werkzeuge und Standards entwickelt, wobei die Magdeburger Medizininformatik eine wesentliche Rolle spielt.

Für die Realisierung des Managed Care (Disease Management, Shared Care) ist die Kommunikation und Kooperation zwischen den Strukturen und Vertretern des Gesundheitswesens basierend auf dem Konsens des informierten Patienten bei gleichzeitiger Wahrung des Vertrauensverhältnisses zwischen Arzt und Patient unverzichtbar. Im Falle der verzahnten Versorgung von Patienten müssen Tausende von Items tausender Patienten zwischen Tausenden von Mitarbeitern des Gesundheitswesens kooperativ kreiert und genutzt werden. Aus datenschutzrechtlichen und ethischen Gründen ist dieser Austausch an die direkten oder indirekten Beiträge der Beteiligten an der Versorgung des Patienten zu binden, was zu unterschiedlichen Rechten und Pflichten bezogen auf diese Informationen führt.

Die entsprechenden Verhältnisse sind hoch dynamisch und können nicht durch einen Administrator unter Verwendung einer multidimensionalen Matrix auf die Personen, Umstände und Prozesse, Informationsobjekte, speziellen Items, Zeitbeschränkungen etc. herunter gebrochen und korrekt verwaltet werden. Deshalb werden auch hier die Methoden der Systemmodellierung für Design, Implementierung und Wartung angewandt, um den Herausforderungen zu entsprechen.

Wie viele andere Länder hat auch die Bundesrepublik ein nationales Programm für die Etablierung einer Gesundheitstelematik-Plattform zur Unterstützung der verzahnten Gesundheitsversorgung gestartet [2, 3]. Diese Plattform kombiniert eine kartenunterstützte Kommunikation seitens des Patienten mit einer netzwerkbasierten Interoperabilität zwischen allen Akteuren des Gesundheitswesens. Für die Patientendatenkarte, die so genannte elektronische Gesundheitskarte, wird eine Mehrzweck-Smartcard benutzt. Sie dient als Versichertenkarte, als Impfausweis, als elektronisches Rezept, zur Speicherung von Verweisen auf Komponenten der elektronischen Patientenakte (Electronic Patient Record – EPR oder Electronic Health Record – EHR) bzw. bezogene Informationen wie Medikamenteninformationen im Netzwerk, sowie als Informationsträger zur Unterstützung von Managed Care und Qualitätssicherung.

Ein speziell geschütztes Fach enthält Informationen, die der Patient vor anderen verbergen möchte. Für den Zugriff auf alle Daten außer den medizinischen Notfalldaten und den aufgedruckten administrativen Informationen ist die Authentifizierung durch den Heilberufeausweis (HBA) erforderlich. Außerdem realisiert die elektronische Gesundheitskarte grundlegende Sicherheitsdienste auf der Basis kryptographischer Algorithmen wie die strenge Authentifizierung, Integritätssi-

Autor: Bernd Blobel
Titel: Rahmenarchitektur und Sicherheitsinfrastruktur der deutschen Gesundheitstelematik-Plattform – die MDA Methodologie
In: Jäckel (Hrsg.) Telemedizinführer Deutschland, Ober-Mörlen, Ausgabe 2005
Seite: 89-96

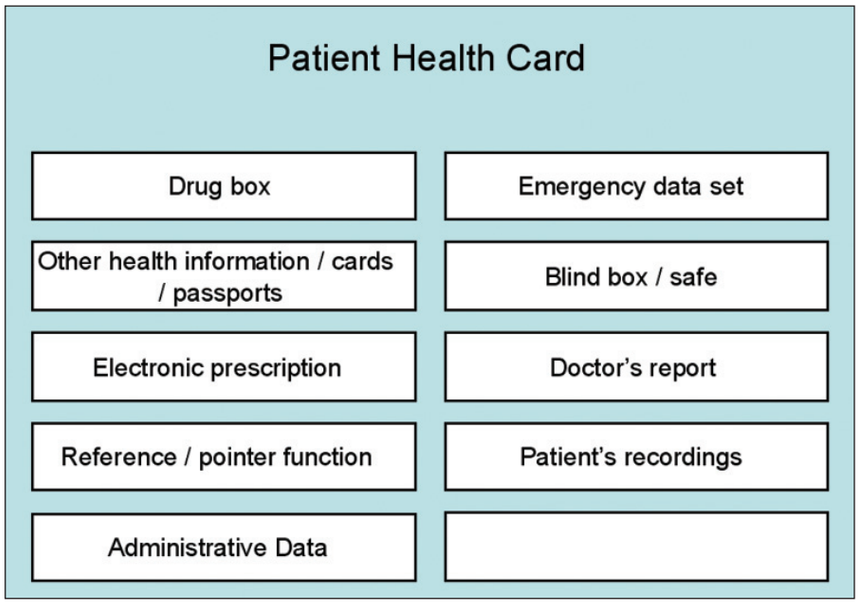

Abbildung 1: *Funktionelle Blöcke der elektronischen Gesundheitskarte*

cherung, Verschlüsselung/Entschlüsselung sowie Verbindlichkeit unter Verwendung der qualifizierten elektronischen Signatur [4] und einer entsprechenden Public-Key-Infrastruktur (Public Key Infrastructure – PKI). Die Gesundheitskarte soll bis Anfang 2006 flächendeckend eingeführt werden. Sie entspricht der künftigen europäischen Versichertenkarte, die bis 2008 in allen Mitgliedsländern der Europäischen Union eingeführt werden soll.

Abbildung 1 zeigt die funktionellen Blöcke der deutschen elektronischen Gesundheitskarte. Die Funktionsblöcke können auf verschiedenen Niveaus getrennt geschützt werden.

Die Sicherheitsdienste unterstützen sowohl Kommunikations- als auch Anwendungssicherheitsdienste für jede Art Principal wie z.B. Nutzer, Geräte, Systeme, Anwendungen, Komponenten oder Objekte. Zur Unterstützung einer vertrauensvollen Interoperabilität zwischen Patienten und Heilberuflern nutzen letztere den HBA zur Realisierung adäquater Sicherheitsdienste.

Um zukunftsfähige Prinzipien für Design und Implementierung von Anwendungen und Sicherheitsdiensten innerhalb der Gesundheitstelematik-Plattform garantieren zu können, wurde vom Bundesministerium für Gesundheit und Soziale Sicherung die Spezifikation einer Rahmenarchitektur mit einer integrierten Sicherheitsinfrastruktur ausgeschrieben [5].

Diese Rahmenarchitektur ist durch folgende Paradigmen charakterisiert:

- Verteilung (auf Internetebene) zur Sicherung der Offenheit,
- Komponentenorientierung für die Skalierbarkeit und Flexibilität,
- Interoperabilität auf Diensteniveau, Konzepte und Wissen mittels formaler Modelle beschreibend,
- Trennung der Plattform unabhängigen und der Plattform spezifischen Modellierung zur Separierung logischer und technologischer Sichten auf die Systemkomponenten sowie
- Installation von Referenz- und Domainmodellen sowie Definition eines abgestimmten Vokabulars zur Sicherung der Interoperabilität.

Die Methodik entspricht völlig der von der Object Management Group (OMG) definierten modellgetriebenen Architektur (Model Driven Architecture – MDA).

Modellierung von Systemen

Um Systeme und ihr Verhalten in vernünftiger Weise zu beschreiben, müssen reale Systeme modelliert werden. Dabei kann das Modell zum einen die interne strukturelle Komplexität eines Systems verbergen (Black Box), zum anderen kann es auf spezielle Aspekte wie z.B. auf seine Form, seine spezielle Funktion oder bestimmte Eigenschaften fokussiert sein. Neben diesem Weg der Vereinfachung eines Systems durch Modellierung können auch Elemente des Systems entsprechend spezifischer Gemeinsamkeiten in Struktur und/oder Funktion gruppiert werden, was das Design, die Entwicklung und Wartung eines Systems verwaltbar, realisierbar und finanzierbar machen kann. Das Ergebnis sind dann Komponenten, die (weitgehend) unabhängig von den anderen gestaltet, hergestellt und verbessert werden können, wobei jedoch eine vernünftige Interoperabilität zwischen bezogenen Komponenten ermöglicht werden muss.

Um die Komplexität des gesamten Gesundheitssystems, welches dem Shared-Care-Paradigma entsprechend aus vielen Subsystemen besteht, zu reduzieren, wird ein einziges, unrealistisch umfassendes Informationssystem, das alle denkbaren Prozeduren, Tatsachen und Ergebnisse abdeckt, durch Subsysteme, die auf spezifische Aufgaben, Inhalte etc. beschränkt sind, realisiert. Mit anderen Worten wird eine Verschiebung vom System zu Komponenten vollzogen.

Ein Informationssystem reflektiert Prozesse in der realen Welt, wobei es einerseits ein informationsbezogenes Modell (z.B. eines Produktionsprozesses) etabliert und andererseits ein reales System (eine Rechneranwendung) implementiert. Modelle sind ebenfalls Systeme, die aus Komponenten bestehen. Das Komponenten-Paradigma ist ein Grundparadigma, das auf reale Systeme sowie auf Modelle der Realität anwendbar ist [1]. Deshalb soll das Komponenten-Paradigma kurz vorgestellt werden.

Methoden

Ein Modell ist eine Beschreibung der Realität unter beschränkten Aspekten oder Constraints. Deshalb spricht man auch von Constraint Models, die unter Verwendung spezieller Modellierungs- und Spezifikationssprachen wie UML und XML beschrieben werden. Diese Sprachen, auch Metasprachen genannt, erlauben nicht nur die Beschreibung der

Modelle, sondern auch die Beschreibung, wie solche Modellbeschreibungen aussehen müssen. Mit Metasprachen können Sprachen spezifiziert werden. Folglich können die Modelle auf unterschiedlichen Meta-Levels sowie auch als Instanzen repräsentiert werden. Die Abstraktion eines Modells gegenüber der Realität kann zum einen die Komplexität der Systeme bzw. ihrer Komponenten im Sinne der Granularität oder Detailliertheit betreffen. Es ist durchaus sinnvoll, Details in einem Modell zu verstecken und sich lediglich auf die groben Zusammenhänge zu beschränken. Zur Funktionsfähigkeit müssen jedoch auch die Details gesichert sein.

Auf der Ebene der höchsten Detailliertheit (Atomic Components) werden die Basiskonzepte einer Ontologie beschrieben. Zur Verringerung der Komplexität einer Fragestellung werden die erforderlichen Komponenten eines verteilten Systems nach den ISO Reference Model – Open Distributed Processing (RM-ODP) aus verschiedenen Blickwinkeln, also in der Fokussierung auf bestimmte interessierende Aspekte modelliert.

Das RM-ODP definiert dabei die fünf Sichten Enterprise View, Information View, Computational View, Engineering View und Technology View [1]. Der Enterprise View beschreibt den Verwendungszweck (Use Cases, Scenarios, Policies) eines ODP Systems und seiner Komponenten. Der Information View betrachtet ihren Inhalt, ihre Funktion und ihre Relationen (Attributes, Operations, Associations) und der Computational View ihre logisch sinnvolle Komposition bzw. Dekomposition. Der Engineering View beschreibt die physische Verteilung des Systems und seiner Komponenten und der Technology View ihre technische Umsetzung (einschließlich Training usw.). Komponenten können zu verschiedenen Domänen gehören, d. h. Funktionen aus verschiedenen Domänen (z. B. Medizin, Finanzwesen, Rechtswesen, Sicherheit) realisieren. Die Domänenmodelle, für die die Komplexität hinsichtlich der Granularität und auch der Fragestellung (Views) wie beschrieben reduziert und die Teilmodelle dann aggregiert werden können, lassen sich zunächst unabhängig voneinander entwickeln.

Views von generischen Komponenten in Gesundheitsinformationssystemen wurden erstmals 1995 beschrieben und 1997 veröffentlicht [2]. Weitere Details finden sich in [3-5] sowie in Abbildung 2.

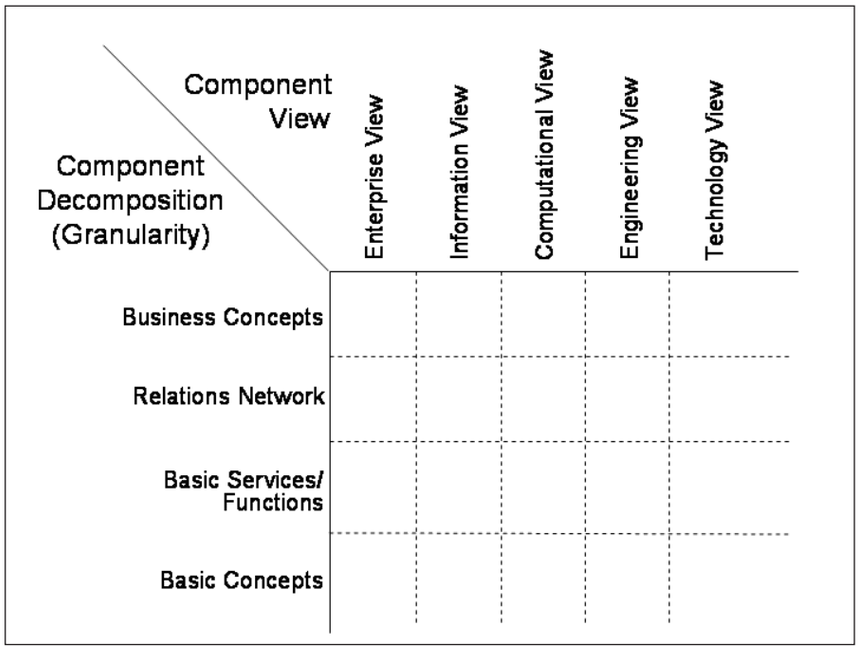

Abbildung 2: *Abstraktionsmatrix von Komponentensystemen*

Modellierung von Sicherheitsdiensten

Während die Modellierung von technischen, medizinischen oder administrativen Funktionen und Prozessen inzwischen weit eingeführt ist, stellt die Formalisierung und Modellierung von Sicherheitsdiensten noch eine Innovation und somit eine besondere Herausforderung dar. Deshalb soll im Folgenden die Modellierung von Sicherheitsdiensten im Mittelpunkt der Betrachtungen stehen.

Datenschutz- und datensicherheitsrelevante Aspekte betreffen die Domänen von Security, Safety und Privacy (da die deutsche Sprache prinzipiell nicht zwischen Security = Sicherheit und Safety = Sicherheit unterscheidet, wurden hier und auch an einigen anderen Stellen die englischen Termini benutzt). Für die verschiedenen Domänen können unterschiedliche Geschäftskonzepte definiert werden. Im Bereich der Datensicherheit beschreiben die Konzepte der Kommunikationssicherheit und der Anwendungssicherheit die entsprechenden Geschäftskonzepte. Kommunikationssicherheit kann über sichere Objekte oder über sichere Kanäle realisiert werden. Security Enhanced EDI (HL7, EDIFACT, xDT, XML) ist ein typisches Beispiel für die erste Alternative, das SSL/TLS Protokoll ist ein Beispiel für die zweite.

Kommunikationssicherheitsdienste sind die wechselseitige starke Identifikation/Authentifikation, die Principal-Zugriffskontrolle, die Sicherung der Integrität, Vertraulichkeit, Verbindlichkeit und Verfügbarkeit der kommunizierten Informationen sowie weitere Notariats-Services. Anwendungssicherheit dient z. B. der Realisierung der Autorisierung und Zugriffskontrolle einschließlich der Definition und dem Management von Rollen und der Entscheidungsunterstützung.

Modellgetriebene Analyse und Design beinhalten die Spezifikation von Anforderungen und Lösungen als erforderliche oder realisierte Services. Diese Services werden durch spezifische Mechanismen implementiert, die verschiedene Algorithmen nutzen, welche auf unterschiedliche Daten angewendet werden.

Privilege Management und Autorisierung können individuellen Akteuren oder Gruppen von Akteuren, die die

gleiche Rolle spielen, zugeordnet werden. Akteure, die mit Systemkomponenten interagieren, werden Principals genannt. Wie bereits eingeführt, können Principals können z. B. Personen, Organisationen, Systeme, Geräte, Anwendungen, Komponenten oder Objekte sein.

Generische Modelle

Für das Privilege Management und die Zugriffskontrolle müssen zwei Basis-Klassentypen behandelt werden:

- Entities
  - Principals
  - Policies
  - Rollen
  - Dokumente

- Aktivitäten (Acts)
  - Policy Management
  - Principal Management
  - Privilege Management
  - Authentifizierung
  - Autorisierung
  - Zugriffskontroll-Management
  - Audit

Die aufgeführten Aktivitäten sind erforderlich, um beschriebenen Sicherheitsservices zu ermöglichen. Im Folgenden wird eine Serie von Modellen beschrieben, die definieren, wie diese Aktivitäten durchgeführt werden.

Auf die folgenden Modelle wird etwas detaillierter eingegangen werden.

- Domänen-Modell
- Policy Modell
- Rollenmodell
- Kontrollmodell
- Delegierungsmodell
- Autorisierungsmodell
- Zugriffskontrollmodell

Alle diese Spezifikationen müssen offen, plattformunabhängig, portabel und skalierbar gehalten werden, um ein breites Spektrum klinischer Bedingungen zu unterstützen und in verschiedenen Ländern mit unterschiedlichen nationalen und berufsständigen Regularien benutzt zu werden. Deshalb werden alle Modelle auf Metamodell-Niveau beschrieben, das Instanzlevel so weit als möglich vermeidend.

Domänen-Modell

Nach gemeinsamen Eigenschaften wie z. B. Policies, Umgebungsbedingungen oder technologischen Lösungen werden Komponenten gruppiert, wobei sie Domänen wie z. B. eine Policy-Domäne, eine Umgebungs-Domäne oder eine Technologie-Domäne bilden. Eine Domäne ist durch einen Domain Identifier, einen Domain Name, eine Domain Authority und einen Domain Qualifier charakterisiert. Eine Sicherheits Policy hat einen Policy Identifier, einen Policy Name und von der Domäne geerbte weitere Attribute.

Policy Model

Eine Security Policy beschreibt den Komplex der rechtlichen, ethischen, sozialen, organisatorischen, psychologischen, funktionellen und technischen Implikationen vertrauenswürdiger Gesundheitsinformationssysteme. Eine Policy ist die Formulierung des Konzeptes der Anforderungen und Bedingungen für die vertrauenswürdige Erzeugung, Speicherung, Verarbeitung, und Verwendung sensitiver Informationen. Eine Policy kann

- verbal, unstrukturiert
- strukturiert unter Verwendung von Schemata oder Templates oder
- formal modelliert

ausgedrückt werden.

Aus Interoperabilitätsgründen muss eine Policy in einer Weise formuliert und kodiert werden, die ihre korrekte Interpretation und Anwendung ermöglicht. Deshalb müssen Policies in Bezug auf Syntax, Semantik, Vokabular und Verarbeitung von Policy-Dokumenten, so genannten Policy Statements oder Policy Agreements beschränkt werden. Ein gemeinhin beschrittener Weg Constraints auszudrücken ist die Spezifikation nutzerdefinierter Schemata wie z.B. XML Schemata. Diese Schemata müssen aus den angesprochenen Interoperabilitätsgründen standardisiert werden.

Um auf eine spezifische Policy verlässlich Bezug nehmen zu können, muss die Policy Instanz einen eindeutigen Namen und einen eindeutigen Policy Identifier haben. Das Gleiche gilt für alle Policy Komponenten wie Domänen, Targets, Operationen und deren bezogene Policies, die ebenfalls eindeutig bezeichnet und identifiziert sein müssen. So wie jede andere Komponente können auch Policy-Komponenten entsprechend dem generischen Komponentenmodell zu Superkomponenten aggregiert oder in Subkomponeten zerlegt werden [3].

Somit ist eine Policy u. a. charakterisiert durch einen Policy Identifier, einen Policy-Name, eine Policy Authority, einen Domain Identifier, einen Domain Name, eine Target List, einen Target Identifier, einen Target Name, das entsprechende Target Object, die zulässigen Operationen und die bezogenen Policies.

Nach [6] kann die Policy-Klasse in Basic Policy, Meta Policy und Composite-Policy-Klassen spezialisiert werden. Die Detailspezifikationen der PONDER Policy Class sind in den Tabellen 1 und 2 beschrieben.

Eine andere Form der Dekomposition von Policies wurde in der Security Services Specification der Object Management Group (OMG) definierte, wobei zwischen folgenden Policies unterschieden wird [7]:

Basic Policy Type	Verwendungszweck
Authorisation Policies	definiert die erlaubten Aktionen
Obligation Policies	sind ereignisgetriggert und definieren Aktionen, die durch Manager-Agenten realisiert werden müssen
Refrain Policies	definiert Aktionen, die ein Subjekt nicht ausführen darf
Delegation Policies	definiert, welche Autorisierungen an wen delegiert werden dürfen

Tabelle 1: *PONDER Basic Policy Types [6]*

Composite Policy Type	Verwendungszweck
Groups	definiert einen Scope für bezogene Policies, auf die ein Set von Constraints angewendet werden kann
Roles	definiert eine Gruppe von Policies (Autorisierungs-, Verpflichtung und Ausschluss-Policies) bezogen auf die Position in einer Organisation
Relationships	definiert eine Gruppe von Policies, Interaktionen zwischen einem Set von Rollen betreffend

Tabelle 2: *PONDER Composite Policy Types [6]*

- Invocation access policy zur Implementierung der Zugriffkontroll-Policy für Objekte
- Invocation audit policy zur Kontrolle von Ereignistypen und Kriterien für das Audit
- Secure invocation policy zur Spezifikation von Sicherheits-Policies, die mit Sicherheitsverbindungen und dem Schutz von Nachrichten verknüpft sind.

Betrachtet man die Anforderungen für unterschiedliche Objekttypen, sind die

- Invocation Delegation Policy,
- Application Access Policy,
- Application Audit Policy und
- Non-Repudiation Policy

zu definieren.

Rollen-Modell

Principals können nach gemeinsamen Rollen, die sie in ihren Beziehungen spielen, gruppiert werden. Für das Etablieren und Verwalten der Relationen zwischen Entities können strukturelle und funktionelle Rollen definiert werden. Jedem Principal können Rollen zugewiesen werden. Da Principals Akteure in Szenarien sind, sind Rollen assoziiert mit Akteuren und mit Aktionen.

Rollen liefern ein Mittel zur indirekten Zuweisung von Privilegien an Individuen. Rollenzuweisungs-Zertifikate werden an Individuen herausgegeben, die ihnen eine oder mehrere Rollen durch Rollenattribute, die in jedem Zertifikat enthalten sind, zuweisen. Spezifische Privilegien werden eher einem Rollennamen über die Rollenspezifikations-Zertifikate zugewiesen als einem individuellen Privilegieninhaber über seine Attributzertifikate. Diese indirekte Zuweisung ermöglicht zum Beispiel, dass Privilegien, die einer Rolle zugewiesen worden sind, aktualisiert werden können, ohne die Zertifikate zu beeinflussen, die Rollen Individuen zuweisen. Rollenzuweisungs-Zertifikate können Attribut-Zertifikate oder Public-Key-Zertifikate sein. Rollenspezifikations-Zertifkate können Attribut-Zertifikate, jedoch nicht Public-Key-Zertifikate sein. Wenn keine Rollenspezifikations-Zertifikate benutzt werden, kann die Zuweisung von Privilegien auf anderem Wege realisiert werden (z. B. lokal konfiguriert als Privilege Verifier).

Wenn das Rollenzuweisungs-Zertifikat ein Attribut-Zertifikat ist, ist das Rollenattribut (Role Attribute) in der Attributkomponente (Attribute Component) des Attribut-Zertifikates enthalten. Wenn das Rollenzuweisungs-Zertifikat ein Public-Key-Zertifikat ist, ist das Rollenattribut in der subjectDirectoryAttributes extension des ID-Zertifikates enthalten. Im letzteren Fall ist jedes Privileg, das im Public-Key-Zertifikat enthalten ist, ein Privileg, das dem Zertifikat-Subjekt (Zertifikat-Halter) zugewiesen wurde, und kein Zertifikat, das einer Rolle zugewiesen wurde.

Wie bereits ausgeführt, können zwei Rollentypen unterschieden werden: strukturelle Rollen und funktionelle Rollen. Strukturelle Rollen reflektieren die strukturellen Aspekte der Beziehungen zwischen Entitäten. Strukturelle Rollen beschreiben Voraussetzungen, Fähigkeiten oder Kompetenzen für die Durchführung von Aktivitäten. Funktionelle Rollen reflektieren die funktionellen Aspekte der Beziehungen zwischen Entitäten. Funktionelle Rollen sind an die Realisierung/Ausführung von Aktionen gebunden.

Modell für funktionelle Rollen

Innerhalb der Geschäftsprozesse des Gesundheitswesens können funktionelle Rollen auf der Ebene der Autorisierungen und Zugriffsrechte in der folgenden generischen Weise definiert werden, wobei die Definitionen aus dem australischen HealthNet-Projektes in leicht veränderter Weise und unter Berücksichtigung anderer Veröffentlichungen verwendet wurden:

- Subject of care (normalerweise der Patient)
- Subject of care agent (Eltern, Vormund, Fürsorge oder andere rechtliche Repräsentanten)
- Responsible (personal) healthcare professional (der Angehörige des Gesundheitswesens, der die engste Beziehung zum Patienten hat, meist der Hausarzt)
- Privileged healthcare professional
- nominated by the subject of care
- nominated by the healthcare facility of care (die Nominierung kann durch Bestimmungen, praktische Erfordernisse, etc. geregelt sein)
- Healthcare professional (direkt einbezogen in die Versorgung des Patienten)
- Health-related professional (indirekt einbezogen in die Versorgung des Patienten, z. B. über Lehre, Forschung, etc.)
- Administrator (und alle anderen Parteien, die in unterstützende Services zur Versorgung des Patienten eingebunden sind)

Modell für strukturelle Rollen

Eine Entity-Entity-Beziehung kann einen Akt der Akkreditierung/Zertifizierung betreffen, bei dem die beteiligten Entitäten spezielle funktionelle Rollen realisieren. Das Zertifikat kann zum Beispiel die Zulassung eines Arztes betreffen, der in diesem Fall der Client ist, während die KV die Certification Authority ist. Das Zertifikat bestätigt die Zulassung zur Ausübung der Praxis auf einem bestimmten Gebiet, welches die entsprechende strukturelle Rolle ist. Ein anderes Beispiel kann sich auf die Approbation als Ergebnis einer Qualifizierung beziehen. Auch hier ist das Ergebnis eine strukturelle Rolle, die des approbierten Arztes.

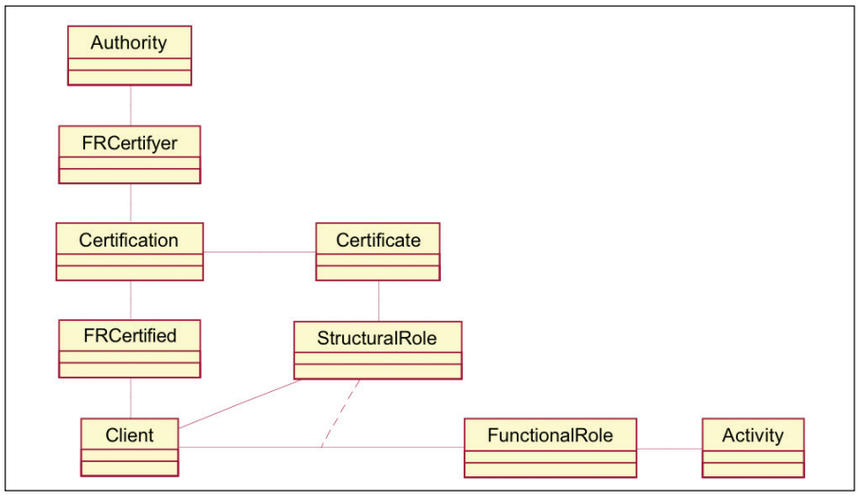

Abbildung 3: *Zuweisung einer strukturellen Rolle innerhalb einer Aktion mit entsprechenden funktionellen Rollen der beteiligten Entitäten*

Diese strukturellen Rollen beschränken bzw. Qualifizieren die Entity-Entity-Beziehungen und können die funktionellen Rollen beeinflussen, die die Entitäten bei der Ausführung einer Aktion spielen. Abbildung 3 zeigt die Zuweisung einer strukturellen Rolle in einer Zertifizierungsaktion mit den entsprechenden funktionellen Rollen der beteiligten Entitäten.

Betrachtet man die strukturellen und die funktionellen Rollen in ihrer Wirkung auf die Entity-Entity-Relationships, beschreiben die strukturellen Rollen Voraussetzungen/Kompetenzen der entsprechenden Entität zur Durchführung einer Aktion, während die funktionellen Rollen die Durchführung der Aktion beschreiben. Qualifikationen, Fähigkeiten und Fertigkeiten beeinflussen sowohl die Zuweisung einer bestimmten strukturellen Rolle als auch die Performanz bei der Aktion innerhalb der funktionellen Rolle.

```
<role>
        <role_name/>
        <role_ID/>
        <role_authority/>
        <authority_ID/>
        <role_description>
                ...
        </role_description>
</role>
```

Abbildung 4: *Generische Rollen-Spezifikation*

Generische Rollen-Spezifikation

Abbildung 4 gibt den Body für eine minimale Rollenspezifikation wieder. Auch hier gelten wie bei der Policy-Definition zusätzliche Beschränkungen z. B. bezüglich der Gültigkeitsdauer einer Rollendefinition.

Danach muss eine Rollendefinition über Namen und Identifier identifizierbar sein, auf die zuständige Authority einschließlich ihres Identifier verweisen und zumindest die Rolle beschreiben. Zusätzlich können weitere administrative Beschränkungen, wie z. B. die Trennung der Verantwortlichkeiten (Vermeidung, das Anweisender zugleich Prüfer ist), fixiert werden.

Das Kontrollmodell

Das Kontrollmodell illustriert, wie die Kontrolle des Zugriffs auf sensitive Objekte realisiert werden kann. Das Modell hat vier Komponenten: den Anfordernden (Claimant), den Prüfer (Verifier), das sensitive Objekt (Target) und die Kontroll-Policy. Umgebungsvariable, wie z. B. der Zeitpunkt oder auch konkurrierende Policies (z. B. Konsens) wirken einschränkend bzw. parametrisierend. Der Anfordernde besitzt Privilege-Attribute (Privilege Attributes), die in Attribute-Zertifikaten (Attribute Certificates) gehalten werden. Das Objekt ist durch Sensitivitäts-Attribute (Sensitivity Attributes) charakterisiert, die in einem Security Label, in einem Attribute-Zertifikat, oder in einer lokalen Datenbank gehalten werden. Die beschriebene Verfahrensweise ermöglicht es dem Prüfer, den Zugriff auf das sensitive Objekt durch den Anfordernden in Übereinstimmung mit der Zugriffs-Policy zu kontrollieren.

Die Privilegien des Anfordernden sind typischerweise im Attribut-Zertifikat eingekapselt. Das kann entweder dem Prüfer bei der Anforderung des Services präsentiert werden (push strategy) oder es kann durch andere Maßnahmen wie z. B. ein Verzeichnis (Directory) verteilt werden (pull strategy). Die Kontroll-Policy muss in ihrer Integrität und Authentizität gesichert werden. Dazu kann sie auch mit den Privilegien des Anfordernden in seinem Attribut-Zertifikat kombiniert werden. Normalerweise wird das Zertifikat in eine separate Struktur überführt.

Der Anfordernde kann eine durch ein Public Key Certificate identifizierte Entität oder ein ausführbares Objekt, welches durch einen Digest identifiziert ist, sein.

Die Allgemeingültigkeit dieses Modells führt dazu, dass die Namen seiner Komponenten etwas abstrakt erscheinen. Mit einer geeigneten Interpretation kann es in allen Situationen angewendet werden.

Delegierungsmodell

Zusätzlich zum Kontrollmodell wird ein Delegierungs-Modell benötigt. Das Delegierungs-Modell hat drei Komponenten: den Prüfer (Verifier), die Autoritäts-Quelle (Source of Authority) und den Anfordernden (Claimant).

Der Prüfer bedient sich einer Autorität, die mit unbegrenzten Privilegien ausgestattet ist. Die Autorität ist ein spezieller Typ einer Attribut-Autorität (Attribute Authority). Sie delegiert durch die Herausgabe von Attribut-Zertifikaten Privilegien an den Anfordernden. Der Anfordernde behauptet seine delegierten Privilegien durch Vorlage seiner Identität. Das kann durch Überprüfung der Kenntnis eines privaten Schlüssels, dessen öffentliches Gegenstück im Public Key Zertifikat enthalten ist,

geschehen. Das Public Key Zertifikat ist durch ein Attribut-Zertifikat referenziert, das seinerseits die angeforderten Privilegien enthält. Im Fall eines ausführbaren Objektes kann dies alternativ durch den Nachweis geschehen, dass der Digest der gleiche wie der des Eigentümerwertes eines Attribut-Zertifikates ist, welches die angeforderten Privilegien einschließt.

Optional kann der Anfordernde seine Privilegien an einen anderen Anfordernden delegieren. Der Prüfer muss bestätigen, dass alle Entitäten im Delegierungs-Pfad ausreichende Privilegien besitzen, um auf das angeforderte Objekt zuzugreifen, welches durch den ursprünglichen Anfordernden angefordert wurde.

Die Autorität kann auch eine Anforderung von einer Entität verarbeiten, um seine Privilegien durch Herausgabe eines Attribut-Zertifikates an eine andere Entität zu delegieren.

Der Anfordernde und der Prüfer können Entitäten aus verschiedenen Sicherheitsdomänen sein. In solchen Fällen kann die Autorität in der Domäne des Prüfers lokalisiert sein, wobei ein durchgehender Teil des Delegierungs-Pfades, der den ursprünglichen Anfordernden enthält, in einer anderen Sicherheits-Domäne sein muss.

Der Delegierungs-Pfad ist verschieden vom Pfad der Validierung des Zertifikates, der benutzt wird, um das Public Key Zertifikat der Entität zu validieren, die in den Delegierungs.Prozess einbezogen ist. Die Qualität der Authentizität, die durch den für das Validierungs-Prozess Public Key Zertifikat angebotenen wird, muss der Sensitivität des zu schützenden Objektes entsprechen.

Für die Spezifizierung der Interoperabilität zwischen Objekten bzw. Komponenten hat die Object Management Group (OMG) in ihrer Security Services Specification ein Delegationsmodell definiert [7]. In einem Objekt-System ruft ein Client ein Objekt zur Durchführung einer bestimmten Operation auf. Dieses Objekt wird die Operation oft nicht vollständig (oder vielleicht gar nicht) ausführen können. Stattdessen ruft es dann ein anderes Objekt zur Erledigung dieser Operation auf. Daraus resultieren in der Regel Ketten von Objekt-Aufrufen.

Die Kommunikation von persönlichen Gesundheitsdaten ist häufig mit einer „Lieferantenkette“, die die Transaktionen ausführt, verbunden. Das betrifft z. B. Sekretariate, Schwestern, Mitarbeiter, Dienstleistungseinrichtungen, oder andere Principals. Das angegebene Delegierungsmodell kann für jegliche Art von Kommunikationsketten benutzt werden.

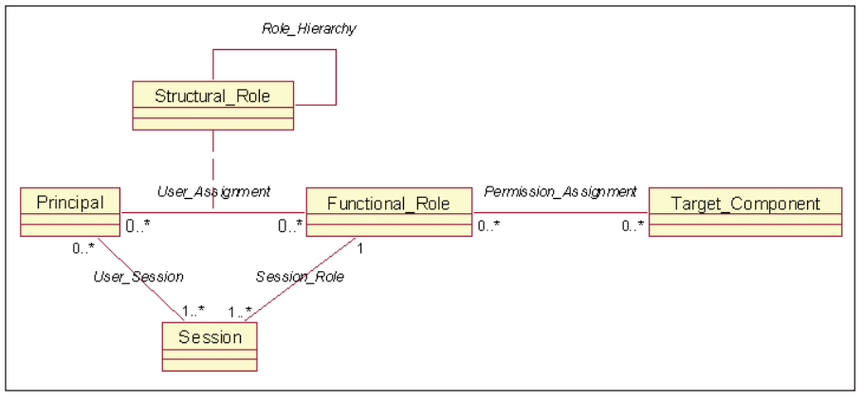

Abbildung 5: *Role-Based Access Control Schema [9]*

Autorisierungsmodell

Das allgemeine Privilege-Management-Modell besteht aus drei Entitäten: dem Informationsobjekt, dem Privilege Asserter und dem Privilege Verifier. Drei prinzipielle Entscheidungen können im Kontext des Privilege Management realisiert werden:

- Anforderung wird autorisiert,
- Anforderung wird angewiesen, und
- Anforderung wird modifiziert.

Credentialing, Privilegierung und Autorisierung werden durch Verknüpfung von Rollen mit Policies realisiert.

Zugriffskontrollmodell

Indem verschiedene Rollenmodelle aus den vorangegangenen Abschnitten und fortgeschrittene Zugriffskontrollmodelle harmonisiert werden, kann folgendes adaptiertes Role-Based Access Control Schema entwickelt werden (Abbildung 5).

Jede Modellkomponente ist definiert durch die Subkomponenten:

- Serien von Basiselementen (Basic Element Sets),
- ein Set von RBAC-Relationen, die die Basiselemente einbeziehen und Subsets von Kartesischen Produkten, die gültige Zuweisungen beschreiben, einschließen, und
- ein Set von Mapping-Funktionen, die zu Instanzen eines Ein-Element-Sets für eine gegebene Instanz eines anderen Element-Sets führen.

Der Zugriff eines Principal auf eine Zielkomponente (Informationsobjekt) wird durch seine funktionelle Rolle im Kontext einer Aufgabe in einem Arbeitsprozess bestimmt, wobei die strukturelle Rolle die Ausführung der Aktion innerhalb der funktionellen Rolle beeinflusst. Die Rechte werden in der jeweils für die Kombination Entität-funktionelle_Rolle-Aufgabe_im_Prozess-Zielkomponente gültigen Policy definiert.

Die Realisierung der MDA im bIT4health-Projekt

Die Ausschreibung des bIT4health-Projektes als Startpunkt für die Spezifikation und Implementierung einer Gesundheitstelematik-Plattform für die Bundesrepublik enthielt klare Vorgaben für die der Rahmenarchitektur und der Sicherheitsinfrastruktur zugrunde zu legenden Prinzipien und Methodologien, die in der Einleitung dieses Beitrages nochmals in Erinnerung gerufen worden sind.

Die letztlich vorgelegten Dokumente zur Spezifikation der Rahmenarchitektur erfüllen nur zum Teil die als K.O.-Kriterium formulierten Forderungen. Das gilt

insbesondere für die Durchgängigkeit der Methodologie sowie für die Modellierung der Sicherheitsdienste. Somit müssen die noch offenen Aufgaben durch die Lösungsarchitektur realisiert werden.

Schlussfolgerungen

Die Entwicklung hochkomplexer, verteilter, intelligenter Informationssysteme ist nur in der konsequenten Verfolgung sich etablierender Basis-Paradigmen möglich. Dazu gehören die strikte Komponentenorientierung, die modellgetriebene Systementwicklung, die Integration von Sicherheitsdiensten und Mechanismen als immanenter Teil der Architektur, die automatische Implementierung der Spezifikationen, u.s.w. Spezifikation, Implementierung und Wartung der Anwendungssysteme folgen dem Paradigma der modellgetriebenen Architektur (Model Driven Architecture – MDA). Deshalb ist die Entwicklung formaler Modelle für die sicherheitsrelevanten Entitäten und Aktionen die entscheidende Voraussetzung für derartige Systeme. Im Beitrag wurden diese Paradigmen demonstriert.

Danksagung

Die Autoren sind den Kollegen der sicherheitsbezogenen Arbeitsgruppen von ISO TC 215 und CEN TC 251 sowie insbesondere den amerikanischen Kollegen vom ASTM E31 und von Department of Defense ePeople Project zu Dank verpflichtet.

Literatur

[1] ISO/IEC 10746 "Information technology – Open Distributed Processing – Reference Model"

[2] Blobel B: Assessment of Middleware Concepts Using a Generic Component Model. Proceedings of the Conference "Toward An Electronic Health Record Europe '97", pp 221-228. 20-23 October 1997 London

[3] Blobel B: Application of the Component Paradigm for Analysis and Design of Advanced Health System Architectures. International Journal of Medical Informatics 60 (3) (2000) pp. 281-301)

[4] Blobel B: Analysis, Design and Implementation of Secure and Interoperable Distributed Health Information Systems. Series Studies in Health Technology and Informatics, Vol. 89. IOS Press, Amsterdam 2002

[5] Blobel B, Pharow P (Edrs.): Advanced Health Telematics and Telemedicine. Volume 96 Studies in Health Technology and Informatics. Amsterdam: IOS Press, 2003

[6] Damianou N, Dulay N, Lupu E, Sloman M. Ponder: A Language for Specifying Security and Management Policies for Distributed Systems. The Language Specification, Version 2.3. Imperial College Research Report DoC 2000/1. 20 October, 2000

[7] OMG Inc: The CORBA Security Specification. Framingham: Object Management Group, Inc., 1997

[8] Castano S, Fugini M, Martella G, Samarati P: Database Security. Addison-Wesley Publishing Company, Wokingham, 1995

[9] Ferraiolo DF, Sandhu R, Gavrila S, Kuhn DR, Chandramouli R: Proposed NIST Standard for Role-Based Access Control. ACM Transactions on Information and system Security, Vol. 4 No. 3, August 2001, pp. 224-274.

Kontakt
Priv.-Doz. Dr. Bernd Blobel
Universitätsklinikum Magdeburg
39120 Magdeburg
Leipziger Str. 44
bernd.blobel@mrz.uni-magdeburg.de