Zugriffskontrolle in Groupware –

Ein nutzerorientierter Ansatz

Dipl. inform. Oliver Stiemerling, Dipl. inform. Markus Won, Dr. rer. nat. Dipl. kfm.Volker Wulf
Universität Bonn
ProSEC - Institut für Informatik III
Römerstr. 164, 53117 Bonn
E-Mail: {os | won | volker}@informatik.uni-bonn.de

Abstrakt: Zugriffskontrollsysteme für kooperative Arbeitsumgebungen zeichnen sich durch eine hohe Komplexität aus. Das führt zu Schwierigkeiten bei der Präsentation und Manipulation von Zugriffskontrollsystemen auf der Ebene der Benutzerschnittstelle. Das Problem wird zudem durch die in kooperativen Arbeitsumgebungen vorherrschende starke Dynamik verschärft. Dieser Beitrag stellt einen nutzerorientierten Ansatz vor, dessen grundlegendes Konzept auf Zugriffsregeln basiert. Das Konzept, bei dem Erlaubnisse oder auch Verbote durch Zugriffsregeln repräsentiert werden, wurde auf Basis einer Feldstudie entwickelt. Die Regeln haben einen Gültigkeitsbereich, der durch Faktoren wie Benutzer, Dokumente, Rollen und Zeitintervalle definiert wird. Wir beschreiben, wie dieser Ansatz implementiert und in einem kommerziellen Groupware-System integriert wurde. Abschließend werden die Resultate einer Evaluation des Prototypen auf Basis der thinking-aloud-Methode diskutiert.

Keywords : Groupware, Zugriffskontrolle, Konfliktmanagement, HCI, Rollen, Endbenutzer

1. Einführung

Im Gegensatz zu klassischen Forschungsbereichen in der Informatik wie die der Informations- oder Betriebssysteme, die sich mit Mehrbenutzer-Systemen beschäftigen, ist im Bereich Groupware die Unterstützung der Kooperation zwischen Benutzern das zentrale Anliegen. Aus diesem Grund müssen verschiedene für die Entwicklung von Groupware-Systemen relevante Aspekte neu überdacht werden. Einer dieser Aspekte ist das Problem der Zugriffskontrolle, das dazu dienen soll die Schutzziele Vertraulichkeit und Integrität (siehe [VoKe83]) zu erreichen. Rannenberg et. al. [Rann97] weisen darauf hin, dass speziell moderne Computersysteme mit ihrer hohen Komplexität ausgefeilte Sicherheitsmechanismen benötigen, um die Anforderungen der Benutzer in Bezug auf Flexibilität zu befriedigen. Das trifft insbesondere auch auf den hier untersuchten Fall der Zugriffsrechte in Groupware zu.

Greif und Sarin [GrSa86] zeigen auf, dass die Zugriffskontrollmechanismen aus den oben genannten klassischen Mehrbenutzer-Systemen im Bereich Groupware nicht flexibel genug sind, um Zusammenarbeit in Gruppen zu verwalten. Sie schlagen deswegen die Entwicklung komplexerer Kontrollmechanismen vor, die auch Faktoren wie die Verwendung von Rollen, abstrakten Operationen (neben lesen und schreiben) und Benutzer-Objekt-Beziehungen (z.B. „ist aktueller Benutzer von“) unterstützen.

Diese Gedanken haben die Ausgestaltung der Zugriffskontrolle in kommerziellen Groupware-Produkten beeinflusst. In der wissenschaftlichen Diskussion wurden sie von verschiedenen Autoren aufgegriffen, die eine Vielzahl von Zugriffskonzepten für Groupware-Systeme, wie beispielsweise den Mehrbenutzer-Editor SUITE [ShDe92] oder auch die web-basierte Groupware-Applikation BSCW [Sikk97], entwickelten. Coulouris und Dolimore [CoDo96] entwickelten ein Zugriffskontrollsystem basierend auf einer Fallstudie, anhand der Bedürfnisse an ein solches Modell aus dem wissenschaftlichen Bereich (kooperative Vorbereitung einer Examensarbeit) herausgearbeitet wurden. Schon Anfang der neunziger Jahre wiesen Ellis, Gibbs und Rein [Elli91] jedoch darauf hin, dass die Verwendung von Konzepten wie Negativrechte (s.u.), Vererbung, Hierarchien oder auch Vertreterregelungen aus Sicht der Benutzer schwierig zu handhaben sei. Deshalb forderten sie, die Gestaltung der Benutzerschnittstelle für solche Zugriffskontrollsysteme genauer zu untersuchen, um auch Endbenutzer in die Lage zu versetzen, diese komplexen Modelle handhaben zu können.

Dewan und Shen [DeSh98] weisen darauf hin, dass Zugriffsstrategien in Gruppenarbeitssystemen häufig von den Benutzern eines Systems selbst festgelegt werden (discretionary access control). Um bestimmte Zugriffsstrategien zu unterbinden oder um Fehlbedienung einzuschränken, halten sie den Einsatz einer Meta-Zugriffskontrolle für wichtig, um Administratoren die Möglichkeit zu geben, die Manipulation der aktuellen Zugriffskontrollstrategie einzuschränken. Diese Meta-Regeln können beispielsweise sichern, dass organisationsinterne Normen in Bezug auf Zugriffsrechte nicht geändert werden können (mandatory access control).

Im POLITeam-Projekt [Kloe95, Wulf97, Crem98, PiWu99] wurde unter diesen Gesichtspunkten eine Erweiterung des Zugriffskontrollsystems für das Groupware-System LINKWORKS von DEC [DEC95] entwickelt, das hier näher vorgestellt werden soll. Im Gegensatz zu anderen Ansätzen steht hier der Benutzer[1] im Vordergrund der Bemühungen. Es geht also nicht primär darum, ein Zugriffskontrollsystem in Bezug auf technische Sicherheitsmechanismen neu zu entwickeln. Stattdessen soll die Verständlichkeit des Zugriffskontrollsystems erhöht werden, d.h., die Entscheidungen, die das Zugriffskontrollsystem unter Berücksichtigung der aktuellen Zugriffskontrollstrategie trifft, sollen verständlich gemacht werden. In diesem Zusammenhang ist ein weiteres Ziel, Manipulationen an der Zugriffskontrollstrategie für Benutzer zu erleichtern. Des Weiteren wurde das bestehende Zugriffskontrollsystem um zwei neue Mechanismen „Transparenz“ und „Aushandlung“ erweitert.

Der Beitrag ist wie folgt aufgebaut: Abschnitt 2 gibt einige für unseren Ansatz grundlegende Definitionen. In Abschnitt 3 werden die Ergebnisse einer empirischen Untersuchung zur Erhebung entsprechender Gestaltungsanforderungen genauer beschrieben. Anschließend wird in Abschnitt 4 das darauf basierende neu entwickelte Zugriffskontrollsystem detailliert dargestellt und in Zusammenhang mit anderen Arbeiten diskutiert. Schließlich soll in Abschnitt 5 die Implementation eines Zugriffskontrollsystems für ein existierendes Groupware-System, das im POLITeam-Projekt eingesetzt wird, beschrieben werden.

2. Grundlegende Definitionen und Beschreibung des Ansatzes

In diesem Abschnitt sollen grundlegende Begriffe wie Zugriffskontrollstrategie und Zugriffskontrollsystem definiert werden. Anschließend wird ein kurzer Überblick über den hier verfolgten Ansatz gegeben.

Zugriffskontrollstrategie

> Eine *Zugriffskontrollstrategie* ist ein Entscheidungsverfahren, das angibt, ob ein Subjekt s in einer bestimmten Situation t eine Operation r auf einem Objekt o ausführen darf.

Definition 1 Zugriffskontrollstrategie

In der Praxis bedeutet das, dass ein Computersystem[2] anhand der vier Parameter (s, r, o, t) entscheidet, wie sich das System verhält. Beispielsweise könnte es sein, dass ein Benutzer A ein Dokument D bearbeiten möchte, das auf dem Schreibtisch seines Kollegen B liegt, der gleichzeitig auch Eigentümer des Dokuments ist. Die Situation könnte hier beispielsweise sein, dass das Dokument in einem bestimmten Ordner liegt. Mithilfe der Zugriffskontrollstrategie und unter Beachtung dieser vier Parameter entscheidet nun das System, ob dem Wunsch des Benutzers A entsprochen wird oder nicht.

> Das *Zugriffskontrollsystem* ist ein Teilsystem des Computersystems, das die Ausführung von Operationen auf im System enthaltene Objekte erlauben oder verbieten kann. Die Entscheidung des Zugriffskontrollsystems hängt dabei von der aktuellen Zugriffskontrollstrategie ab.

Definition 2 Zugriffskontrollsystem

Diese Definition impliziert, dass das Zugriffskontrollsystem auf eine Repräsentation der Zugriffskontrollstrategie zugreifen können muss. Diese Repräsentation kann entweder in der Software fest integriert oder aber durch Systemadministratoren oder gewöhnliche Nutzer veränderbar abgespeichert sein. In diesem Aufsatz sollen die durch Benutzer änderbaren Zugriffskontrollstrategien und deren technische Realisierung untersucht werden.

[1] Mit Benutzern (auch Endbenutzern) ist in diesem Zusammenhang ein DV-Anwender mit geringen bis durchschnittlichen Kenntnissen gemeint.

[2] Wir verwenden hier den Begriff Computersystem synonym mit dem Begriff Informationssystem.

Ein konzeptionelles Modell für änderbare Zugriffskontrollen

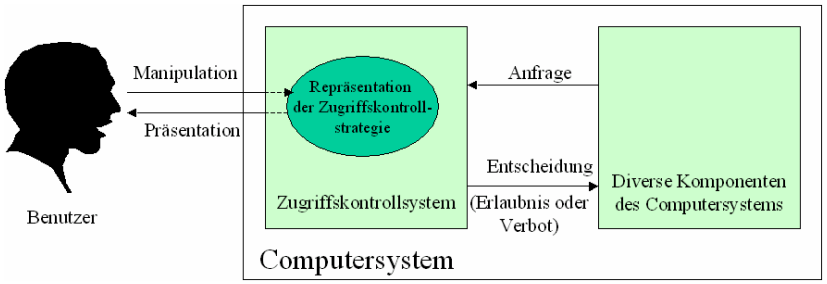

Bild 1 Konzeptionelles Modell für durch Benutzer änderbare Zugriffskontrollen

In Bild 1 ist ein einfaches Modell für eine änderbare Zugriffskontrollstrategie dargestellt. Das Zugriffskontrollsystem kommuniziert mit anderen Komponenten des Computersystems und entscheidet bei entsprechenden Anfragen, ob eine Operation in einer bestimmten Situation durchgeführt werden darf oder nicht. In diesem Fall wird die Repräsentation der aktuellen Zugriffsstrategie ausgewertet, die dem Zugriffskontrollsystem verfügbar sein muss. Da diese vom Benutzer änderbar sein soll, verfügt das Zugriffskontrollsystem über eine Benutzerschnittstelle, die Präsentationsmöglichkeiten zur Verfügung stellt. Weiterhin dient diese Benutzerschnittstelle hier auch zur Manipulation der aktuellen Zugriffskontrollstrategie.

Probleme und Anforderungen

Wie oben angesprochen hat das Zugriffskontrollsystem neben der Entscheidungsaufgabe, ob eine Operation durchgeführt werden darf oder nicht, auch noch die Aufgabe, dem Benutzer die Zugriffskontrollstrategie sowohl zur Ansicht als auch (in änderbaren Zugriffskontrollen) zur Manipulation zugänglich zu machen. Dies führt dazu, dass man die Anforderungen an das Design der Benutzerschnittstelle eines Zugriffskontrollsystems in drei grobe Bereiche wie folgt aufteilen kann:

- ***Repräsentation*:** Im Zugriffskontrollsystem muss jede in den für Computersysteme relevanten Bereichen eingesetzte Zugriffskontrollstrategie darstellbar sein.
- ***Präsentation*:** Die aktuelle Zugriffskontrollstrategie ist dem Benutzer so verfügbar zu machen, dass es ihm möglich ist,
    1. die Entscheidungen des Zugriffskontrollsystems (also die dahinterliegende Strategie) zu verstehen und er
    2. leicht die Unterschiede zwischen der tatsächlichen Zugriffskontrollstrategie und der von ihm intendierten erkennen kann.
- ***Manipulation*:** Das Zugriffskontrollsystem ist so zu gestalten, dass Manipulationen an der Zugriffskontrollstrategie möglichst einfach, effizient und effektiv durchzuführen sind, da üblicherweise solche Strategien nicht ausschließlich von der Organisation festgelegt, sondern im Arbeitskontext von allen Benutzern verändert und erweitert werden (siehe Fallbeispiel weiter unten).

In verschiedenen Arbeiten wurden Ansätze entwickelt, wie man Zugriffskontrollsysteme konstruieren kann, die mächtig genug sind, um Zugriffskontrollstrategien für Mehrbenutzer-Systeme zu verwalten. In den meisten Fällen spielte dabei das Design der Benutzerschnittstelle eine eher untergeordnete Rolle und wurde erst später in das bestehende System integriert. An einem einfachen Beispiel wollen wir nun versuchen zu erläutern, warum die drei oben angesprochenen Aspekte, die bei der Entwicklung eines Zugriffskontrollsystems eine Rolle spielen, nicht getrennt voneinander betrachtet werden dürfen.

Im klassischen Matrix-Modell [Lamp74] verwaltet das Zugriffskontrollsystem eine Matrix M, deren Dimensionen durch die Benutzer und die im System vorhandenen Objekte definiert sind. In jedem Matrix-Element sind also die Zugriffsrechte bez. eines Objektes und eines Benutzers gespeichert. Eine Zugriffsstrategie ist auf diese Weise sehr leicht auszuwerten. Das Zugriffskontrollsystem muss lediglich im passenden Matrix-Element nachsehen, ob eine Operation erlaubt oder verboten ist.

Anhand eines Beispiels für eine einfache Konvention oder Regel, wie sie in Organisationen häufig gilt, lassen sich zwei Probleme des klassischen Matrix-Modells leicht veranschaulichen.

Soll die Regel „Benutzer der Abteilung D dürfen alle Objekte benutzen, die auf meinem Schreibtisch liegen“ einer aktuellen Zugriffsstrategie hinzugefügt werden, muss die gesamte Matrix durchsucht werden. Es ist nötig, für jeden Benutzer zu prüfen, ob er zur Abteilung D gehört oder nicht. Weiterhin muss auch für jedes Objekt die Objektlage (auf welchem Schreibtisch liegt es) ausgewertet werden. Anschließend wird nun in jedem betroffenen Matrix-Element das Leserecht eingetragen. Eine Aufgabe beim Design des Zugriffskontrollsystems

muss also sein, eine Benutzerschnittstelle zur *Manipulation* zur Verfügung zu stellen, die es Benutzern erlaubt, diese Manipulation mit effizienten Methoden durchzuführen.

Außerdem stellt sich das Problem, wie einem anderen Benutzer diese dann in die Zugriffsstrategie integrierte Zugriffsregel *präsentiert* werden kann. Für das System ist es unmöglich, eine solche Zugriffsregel, die über die gesamte Matrix verteilte Einträge induziert, zu erkennen, d.h. herauszufinden, warum einige der Zellen den Eintrag „zum Lesen freigegeben" enthalten. Das System kann dem Benutzer also nicht genau (bez. der ursprünglichen Strategie) erläutern, warum er auf ein Objekt zugreifen kann oder nicht. Diese Informationen (dass der Inhalt einer Zelle das Resultat einer bestimmten Strategie ist) gehen im Matrix-Modell verloren[3].

Das klassische Matrix-Modell kann mittels Zugriffskontrolllisten (Access Control Lists, [Saty90]) oder Berechtigungslisten (Capability Lists [Wulf74]) implementiert werden. Diese Verfahren erweitern nicht die Mächtigkeit des Matrix-Modells, sondern erhöhen lediglich die Effizienz auf technischer Seite. Wie oben beschrieben bleibt jedoch das Problem, Benutzern die Zugriffskontrollstrategie nicht nur zu präsentieren („die gewünschte Aktion ist verboten") sondern auch verständlich zu machen („die gewünschte Aktion ist wegen folgender Regel „.." verboten"). Shen und Dewan [ShDe92] verwenden Vererbungsmechanismen über Hierarchien von Subjekt-, Objekt- und Rechtegruppen, um die Spezifikation von Rechten zu erleichtern. Allerdings sind auch vererbte Rechte über mehrere Hierarchieebenen hinweg zu präsentieren. Um das Verhalten des Zugriffskontrollsystems zu verstehen und zu manipulieren, muss der Benutzer die Struktur der Hierarchien und die damit verbunden Vererbungsmechanismen genau verstehen. Sikkel [Sikk97] vererbte Zugriffsrechte ebenfalls über eine Gruppenhierarchie. Auch dieser Ansatz hat deswegen Probleme bei der Präsentation der Strategie an der Benutzerschnittstelle. Coulouris und Dollimore [CoDo95] verwenden die Repräsentation auf Matrix-Basis. Sie fügen dem grundlegenden Modell das Konzept der Rollen hinzu. Auf diese Weise ist ihr Ansatz zwar wesentlich flexibler in Bezug auf die Zugriffskontrollstrategien, die oben angesprochenen Kritikpunkte am Basis-Matrix-Modell bleiben jedoch unberührt.

Das grundsätzliche Modell der Matrix – auch wenn es in einigen Arbeiten beispielsweise durch Funktionen als Matrix-Einträge erweitert wird – ist all diesen Ansätzen gemein. Erweiterungen existieren aber auch dahingehend, dass man Regelmengen, die Zugriffsrechte definieren, in Form von Funktionen bzw. Relationen festlegt [FSW81]. Diese sind dann zur Laufzeit vom Zugriffskontrollsystem zu prüfen. Auf diese Weise werden die Zugriffsrechte, die sonst in den Matrix-Elementen stehen, on demand generiert. Diese Verfahren werden häufig eingesetzt, um die Flexibilität des Zugriffskontrollsystems zu erhöhen.

In dem hier präsentierten Ansatz wird ein ähnliches Verfahren verwendet. Allerdings geht es darum, die Beschreibung der Regelmengen möglichst dicht an der natürlichen Sprache zu orientieren, um Endbenutzern das Verständnis für die aktuelle Zugriffskontrollstrategie zu erleichtern und auch Änderungen einfach durchführen zu können.

3. Empirische Untersuchungen

Wir haben uns mit Anforderungen an die Gestaltung des Zugriffskontrollsystems in Groupware im Rahmen des POLITeam-Projekts beschäftigt. Um herauszufinden, welche Zugriffskontrollstrategien tatsächlich beim Einsatz von Groupware-Systemen verwendet werden, wurde zunächst ein Workshop veranstaltet. Um zusätzliche Anforderungen bezüglich der zu realisierenden Zugriffskontrollstrategien zu erheben, wurden in drei verschiedenen Anwendungsfeldern semi-strukturierte Interviews durchgeführt. Dabei war für uns wichtig, dass die Interviewpartner aus verschiedenen hierarchischen Ebenen ihrer Organisationen stammen, weil wir aufgrund unserer bisherigen Erfahrungen vermuteten, dass die hierarchische Position einen Einfluss auf die Anforderungen an die Zugriffsrechtvergabe hat.
Unser Ansatz basiert auf der Idee, die Repräsentation der Zugriffskontrollstrategien im System möglichst dicht an Formulierungen durchschnittlich kompetenter DV-Anwender und an deren Sprache zu orientieren. Dies führt dann zu einem Modell, dessen Benutzerschnittstelle für den Anwender einfach zu handhaben ist, sowohl was die Präsentation angeht, als auch wie Manipulationen an der aktuellen Strategie vorzunehmen sind. Daraus folgt, dass in den Interviews nicht nur die Semantik einer Strategie die Art, wie diese Strategie von den Benutzern beschrieben wurde, im Mittelpunkt des Interesses stand.

Beispiele für Zugriffskontrollstrategien

Im Folgenden soll anhand eines der drei untersuchten Beispiel-Anwendungsgebiete kurz erläutert werden, wie Zugriffskontrollstrategien in der Praxis konfiguriert werden.

Die Beispiele und Ergebnisse beziehen sich auf Interviews, die mit Mitarbeitern eines mittelständischen Softwarehauses (ca. 70 Mitarbeiter) geführt wurden. Die Interviews wurden mittels Tonband aufgezeichnet,

[3] Das Matrix-Modell hat weitere Einschränkungen. Es gibt Zugriffskontrollstrategien (z.B. die so genannte „chinese wall"-Strategie), die nicht ausgedrückt werden können [Mins93].

anschließend transkribiert und inhaltsanalytisch ausgewertet. Die Befragten gehören nicht zu den Entwicklern, sondern zur Buchhaltung bzw. zur Geschäftsführung, haben jedoch mehrjährige EDV-Erfahrung aus Anwendersicht. Aus dieser Sicht kennen sie auch die sich aus ihrer Arbeit ergebenden Sicherheitsbedürfnisse im Umgang mit der EDV. Die Buchhaltung der Firma bestand aus einem Geschäftsführer, einer Assistentin, drei Buchhalterinnen und einer studentischen Aushilfskraft. Die drei Buchhalterinnen waren in erster Linie dafür zuständig, Rechnungen auszustellen, neue Rechnungen zu schreiben und die Bücher zu führen. Ein Teil der Abrechnungsaufgaben wurde von einem unabhängigen Steuerberater übernommen. Die studentische Aushilfskraft war dafür zuständig, bezahlte Rechnungen zu summieren oder aber auf noch nicht bezahlte Rechnungen aufmerksam zu machen. Alle diese Daten wurden auf einem Server, der unter Windows NT lief, gespeichert. Die Buchhalterinnen beschrieben im Verlauf der Interviews folgende Zugriffskontrollstrategie: „Zwischen uns dreien gibt es keinen Unterschied. Jede von uns kann auf die Daten der anderen zugreifen, falls sie krank sind oder Urlaub haben. Die Aushilfskraft darf alle Rechnungen des letzten Jahres lesen. Der Geschäftsführer darf alle Daten lesen, jedoch nicht ändern.“ Die letzte Aussage bezieht sich auf einen Vorfall in der Vergangenheit. Der Geschäftsführer änderte damals den Inhalt eines Dokuments, ohne jemanden davon in Kenntnis zu setzen. Dadurch kam es zu größeren Komplikationen und Inkonsistenzen.

Dieses Beispiel zeigt auch, dass die Interviewpartner einige Informationen implizit annahmen. Die oben beschriebenen Regeln lassen keine Aussage darüber zu, welche Zugriffsrechte andere Mitarbeiter hatten, die nicht zur Buchhaltung gehörten. Implizit wurde angenommen, dass alles, was nicht explizit erlaubt ist, verboten ist.

Formulierung der Zugriffskontrollstrategien

Bei der Auswertung der Interviews war es aufgrund der individuellen Unterschiede in Sprache und Ausdruck relativ aufwendig, allgemeine Prinzipien herauszuarbeiten, wie Zugriffskontrollstrategien formuliert werden. Selbstverständlich wichen einige geäußerte Zugriffsregeln von diesen Prinzipien ab. Im Folgenden sollen diese Grundprinzipien jedoch komprimiert beschrieben werden (für eine detaillierte Diskussion der angewendeten Methoden und Resultate siehe [Stie96]).

Beobachtung 1: Zugriffskontrollstrategien werden als Mengen von Erlaubnissen und Verboten ausgedrückt.

In der Regel gaben die Interviewpartner einfache Aussagen an, die ein Verbot oder eine Erlaubnis ausdrücken. Anhand des oben angegebenen Beispiels lässt sich diese Beobachtung leicht nachvollziehen:

1. Mitglieder des Buchhaltung dürfen Rechnungen lesen und schreiben.
2. Kurt (der Geschäftsführer) darf das Hauptbuch lesen.
3. Kurt darf Rechnungen nicht ändern.

Während die ersten beiden Aussagen eine Erlaubnis darstellen, drückt die dritte ein Verbot aus. Dies zeigt die Notwendigkeit der Verwendung „negativer Rechte“ auf, wie auch schon in anderen Arbeiten (siehe auch [ShDe92]) gefordert wurde. Die Mächtigkeit des Zugriffskontrollsystems in Bezug auf Aussagekraft wird dadurch nicht erhöht, wohl aber die Bedienung erleichtert.

Eine Zugriffskontrollstrategie besteht üblicherweise aus einer Menge solcher Aussagen. Einige dieser Aussagen, wie z.B. Regel 1, sind sehr allgemein gehalten (viele Benutzer, mehrere Dokumente), andere sind sehr viel spezieller. So bezieht sich Regel 2 beispielsweise lediglich auf einen Benutzer und eine Dokument.

Beobachtung 2: Zugriffskontrollstrategien werden durch das Aufstellen von Ausnahmen bez. allgemeinerer Regeln verfeinert.

Einige Benutzer beschrieben Zugriffskontrollstrategien dadurch, dass sie zuerst eine sehr allgemeine Regel aufstellten (Regel 1), die sie anschließend durch zusätzliche Regeln verfeinerten. Dabei überlappten sich Regeln, von denen eine ein Verbot, eine andere eine Erlaubnis darstellt, normalerweise nicht. Hier gab es zwei verschiedene Möglichkeiten: Entweder wurde ein generelles Verbot („Niemand darf meine Dokumente lesen“) durch eine spezielle Erlaubnis („Benutzer A darf meine Dokumente lesen“) verfeinert oder umgekehrt.

Dieser Ansatz, Zugriffskontrollstrategien zu beschreiben, führt zu einer einfachen Lösung, wie eine Menge von Erlaubnissen und Verboten zu interpretieren ist. Die Regeln lassen sich so nach dem Prinzip „die speziellste Regel gilt“ auswerten.

Eine weitere Beobachtung war, dass sehr allgemeine Regeln („Alles ist verboten.“ oder „Alles ist erlaubt“) nicht explizit genannt, sondern implizit vorausgesetzt wurden. Allgemein lässt sich sagen, dass die meisten Interviewpartner eher die Regel „Alles ist verboten“ implizit annahmen. Diese implizite Regel wird üblicherweise im Zusammenhang mit der Regel „Der Besitzer eines Dokuments darf auf diesem alle Operationen ausführen“ kombiniert.

Bereich einer Regel

Die Regeln drücken eine Erlaubnis oder ein Verbot für einen bestimmten Bereich aus. Dabei bestimmt die „Größe“ des Bereichs die Generalität. Im Folgenden beschreiben wir, wie die Benutzer diesen Bereich bestimmen, ungeachtet dessen, ob ein Computersystem diese Faktoren bei der Auswertung berücksichtigen kann.

Benutzer und Dokumente
Normalerweise enthalten die Regeln eine Beschreibung der Benutzer und der Dokumente, auf die sie sich beziehen. Sowohl Benutzer als auch Dokumente werden hierbei entweder explizit genannt („Benutzer B darf das Hauptbuch nicht lesen“) oder aber allgemeiner beschrieben („Alle Benutzer der Verwaltung dürfen Rechnungen lesen“). Gruppierungen werden bei den Benutzern anhand von Organisationsstrukturen, Rollen oder individuellen Kriterien („Meine Freunde“ etc.) vorgenommen, Dokumente werden durch ihren Objekttyp, ihren Inhalt oder auch ihre Objektlage („auf meinem Schreibtisch“) beschrieben.

Beziehungen zwischen Benutzern und Dokumenten
In einigen Regeln wird der Bereich dadurch definiert, dass eine Beziehung zwischen Benutzer und Dokument wie z.B. „aktueller Benutzer“- oder „Eigentümer“-Beziehungen beschrieben wird (siehe auch [GrSa86]). In LINKWORKS ist es beispielsweise auch möglich, ein Dokument zu signieren. Dadurch würde auch eine Beziehung wie „gezeichnet von Benutzer A“ möglich.

Weitere den Bereich bestimmende Faktoren
In einigen Regeln wird der Bereich durch eine Zeitangabe eingeschränkt. Diese Zeitangaben können sowohl regelmäßige Form („an Wochentagen“) oder auch absolute Form („4.7.1996 – 6.7.1996“) haben. Weiterhin können sich die Regeln auch auf Operationsgruppen beziehen („Benutzer C darf *alles* mit Dokument F machen“).

Zusammenfassung der Ergebnisse

Die Hauptresultate des ersten Teils der Feldstudie waren folgende Beobachtungen:

- Benutzer formulieren Zugriffskontrollstrategien üblicherweise als Menge von Erlaubnissen und Verboten.
- Diese Regeln gelten für bestimmte Bereiche, die mehr oder weniger weit gefasst sind. Überlappen sich gegensätzliche Regeln, so wird nach dem Prinzip „die speziellste Regel gilt“ verfahren. Dies gibt uns einen ersten Ansatz, wie das Design eines Zugriffskontrollsystems aussehen könnte, das die effektive und benutzerfreundliche Spezifikation von Zugriffskontrollstrategien und deren Manipulation ermöglicht.

Im zweiten Teil der Feldstudie ging es darum herauszufinden, durch welche Faktoren der Bereich, für den eine Regel gilt, definiert wird. In der folgenden Tabelle werden die von den Befragten genannten Faktoren übersichtsartig dargestellt.

Faktor	Beschreibung
Name des Dokuments	Wie heißt das Dokument?
Inhalt des Dokuments	Enthält das Dokument eine bestimmte Zeichenkette?
Entwicklungsstand des Dokuments	In welchem Arbeitszustand (fertig, in Arbeit, korrigiert etc.) befindet sich das Dokument?
Objektlage	Wo im System liegt das Dokument? (eigener Schreibtisch, Verzeichnis X etc.)
Elektronische Signatur	Wurde das Dokument abgezeichnet? Von wem?
Benutzername	Wie heißt der aktuelle Benutzer?
Benutzerrolle	Welche Rolle (Chef, Gruppenleiter etc.) nimmt der aktuelle Benutzer ein?
Organisationseinheit	Zu welcher Organisationseinheit (Verwaltung, Technik, Vorstand etc.) gehört der aktuelle Benutzer?
Gruppenzugehörigkeit des Benutzers	Gehört der Benutzer zu einer bestimmten Gruppe? Zu welcher?
Zeitangaben	Zu welchem Zeitpunkt oder innerhalb welchen Zeitraums möchte der Benutzer die Operation auf dem Dokument ausführen?
Operation	Welche Operation soll auf dem Dokument ausgeführt werden?
Gruppen von Operationen	Zu welcher Gruppe von Operationen gehört die auszuführende Operation?

Tabelle 1 Faktoren, die den Gültigkeitsbereich einer Regel bestimmen

4. Ein regelbasiertes Zugriffskontrollsystem

Im letzten Abschnitt wurden die Ergebnisse der Feldstudie beschrieben. Darin beschrieben Benutzer, wie sie Zugriffskontrollstrategien formulieren und welche Faktoren den Gültigkeitsbereich einer Regel bestimmen können. Im Folgenden soll nun eine Zugriffskontrolle vorgestellt werden, das auf diesen Resultaten basiert. Zuerst sollen die Konzepte des Modells und die Evaluationsstrategie beschrieben und anhand eines Beispiels erläutert werden. Anschließend wird das Problem behandelt, wie mit Regeln, die sich auf das Manipulieren von Regeln beziehen, umgegangen werden soll.

Grundlegende Konzepte

Die zentralen Elemente unseres Zugriffskontrollmodells sind Zugriffsregeln, die Erlaubnisse oder Verbote für einen bestimmten Bereich darstellen.

> Eine *Zugriffsregel* ist ein Paar (s, d), wobei s eine Konjunktion von Prädikaten über die aktuelle Situation ist, $d \in \{\text{erlaubt, verboten}\}$[4] ist ein Flag, das den Typ der Regel angibt. (s steht für scope, d für decision.)

Definition 3 Zugriffsregel

Der Gültigkeitsbereich s

Die möglichen Prädikate des Gültigkeitsbereichs s einer Zugriffsregel (s,d) sind die Systemrepräsentationen der in Tabelle 1 aufgelisteten Faktoren. S könnte beispielsweise folgende Form annehmen:

Benutzer(„Benutzer A“) $\wedge$ Dokument(„Text C“) $\wedge$ Operation(„lesen“).

Eine Regel ist immer dann anwendbar, wenn der aktuelle Zustand des Systems den Gültigkeitsbereich s erfüllt. In diesem Fall ist s erfüllt genau dann, wenn der Benutzer A den Text C lesen möchte. Tabelle 2 zeigt diejenigen Faktoren, die bei der Implementation unseres Modells im POLITeam-Projekt verwendet wurden.[5] Die Reihenfolge der Faktoren besagt, wie wichtig ein Faktor für die Bereichsangabe ist (beispielsweise ist eine Regel, deren Gültigkeitsbereich ein spezielles Dokument enthält, spezieller und hat damit höhere Priorität als eine Regel, deren Bereich eine Objektklasse enthält). Einige der Prioritätseinstufungen entstanden aus Ergebnissen der Feldstudie, z.B. dass Regeln, die sich auf Dokumente beziehen, wichtiger sind als solche, die sich auf Benutzer beziehen (siehe auch [Stie96]).

Faktor	System-Repräsentation
1. Dokumentname	LINKWORKS: Objekt-ID
2. Dokumentklasse	Texte, Graphiken etc.
3. Objektlage	Benutzerschreibtisch, spezieller Ordner etc.
4. Benutzer	LINKWORKS: Benutzer-ID
5. Benutzergruppe	Rollen, Organisationseinheiten, selbstgestaltete Gruppen (Freunde etc.)
6. Zeit/Datum	Erstellungsdatum, letzter Zugriff, letzte Änderung etc.
7. Benutzer-Dokument-Beziehungen	Eigentümer, Ersteller etc.
8. Signatur	LINKWORKS unterstützt das Abzeichnen von Dokumenten.
9. Operation	LINKWORKS: lesen, bearbeiten, löschen etc.

Tabelle 2 Implementierte Faktoren nach Priorität sortiert

Alle Prädikate des Gültigkeitsbereichs s einer Regel werden immer bezüglich des aktuellen Zustands des Systems ausgewertet, d.h. beispielsweise, dass bei der Überprüfung des obigen Beispiels der Name des aktuell ausgewählten Dokuments mit „Text C“ verglichen wird. Im Folgenden wird der Name des Prädikats (Dokumentenname) mit *Prädikat*, der Ausdruck in Klammern (Text C) mit *Argument* bezeichnet. Ein Prädikat p ist enthalten in einem Gültigkeitsbereich s ($p \in s$), genau dann, wenn das Prädikat p Teil der Konjunktion ist, die den Gültigkeitsbereich s beschreibt (z.B. $s = ... \wedge p(x) \wedge ...$).

Die Entscheidung d

Der Parameter d einer Regel (s,d) bestimmt, wie das Zugriffskontrollsystem auf einen Zugriff in einer Situation, die den Gültigkeitsbereich s erfüllt, reagieren soll.

Einfacher Ansatz

Im einfachen Fall kann d genau einen der beiden Werte „erlaubt“ und „verboten“ annehmen. Das Zugriffskontrollsystem arbeitet in einem solchen Fall folgende Schritte ab (siehe Abb. 2):

[4] Die hier aufgeführte Menge der Werte, die d annehmen kann, wird weiter unten noch erweitert.

[5] Einige der in Tabelle 2 aufgeführten Faktoren haben keine direkte Systemrepräsentation.

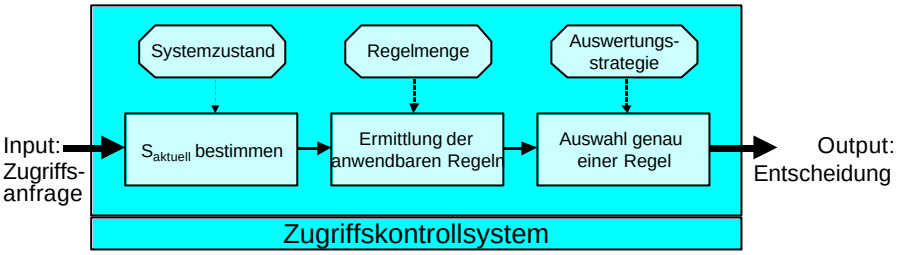

Bild 2 Entscheidung des Zugriffskontrollsystems

Zuerst wird die Situation, in der der Zugriff stattfinden soll, überprüft und ausgewertet. Daraus ergibt sich $s_{aktuell}$. Anschließend wird die Regelmenge durch Vergleich von $s_{aktuell}$ und den jeweiligen Bereichen der Regeln dahingehend untersucht, welche der Regeln in $s_{aktuell}$ anwendbar sind. Durch eine weiter unten beschriebene Strategie wird jetzt genau eine der anwendbaren Regeln ausgewählt. Diese Regel (s,d) bestimmt nun die Reaktion des Zugriffskontrollsystems, welches in diesem Fall lediglich den Parameter d der Regel zurückliefert. In Bild 3 sieht man noch einmal explizit die Aktionsabfolge bei einem Datenzugriff in einer Petri-Netz-Darstellung:

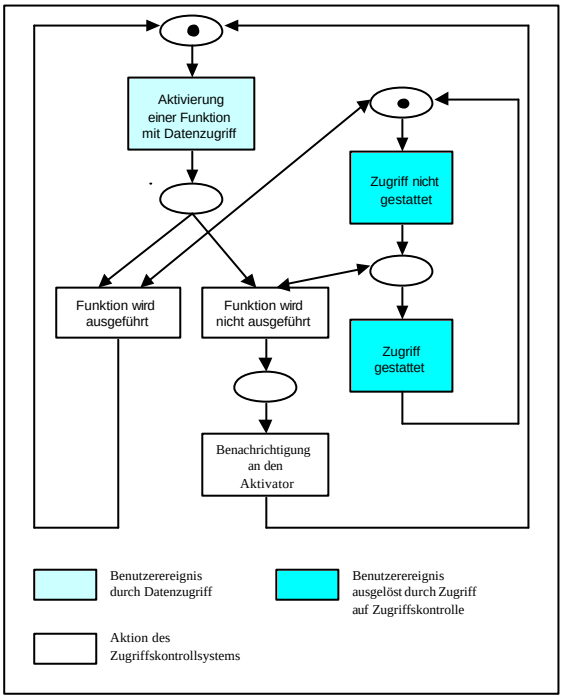

Bild 3 Der einfache Ansatz

Der hier beschriebene Ansatz unterscheidet sich von Zugriffskontrollsystemen der derzeit auf dem Markt befindlichen Systeme wie beispielsweise UNIX [Neme97] oder auch Windows NT [Micr97] u.a. durch die Erweiterung der Entscheidung d um Verbote. Vereinfacht dargestellt ist in den oben angesprochenen Systemen die Standardeinstellung für die Zugriffsstrategie, dass der Besitzer eines Dokumentes auf dieses Vollzugriff hat, wohingegen für alle anderen alle Zugriffe verboten sind. Diese Zugriffskontrollstrategie kann durch die Benutzer (in diesem Fall jeweils der Eigentümer eines Dokumentes) durch explizite Erlaubnisse verändert werden.

Der hier beschriebene Ansatz, als Entscheidung einer Regel auch Verbote zuzulassen, erweitert nicht die Mächtigkeit, Zugriffsregeln zu formulieren. Tatsächlich kann jede Strategie, die mit einer Menge von Erlaubnissen und Verboten definiert ist, auch ausschließlich mit Erlaubnissen formuliert werden, wenn grundsätzlich für alle alles verboten ist. Da es in diesem Aufsatz jedoch um benutzergerechte Repräsentation von Zugriffsstrategien geht, sind Verbote eine Erweiterung, die sowohl die Repräsentation als auch die Manipulation von Zugriffsstrategien stark vereinfacht. Eine Zugriffskontrollstrategie könnte beispielsweise vorsehen, dass alle Benutzer außer „Benutzer X“ alle im System vorhandenen Dokumente lesen dürfen. Wird die Zugriffsstrategie mit Erlaubnissen und Verboten formuliert, so würde man allen das Lesen erlauben und „Benutzer X“ das Lesen

explizit verbieten. Hat man nur Erlaubnisse zur Verfügung, so müsste man jedem Benutzer außer „Benutzer X“ das Lesen separat erlauben[6].

Die Menge der Werte, die d annehmen kann, lässt sich nun auf verschiedene Arten erweitern. Pfeifer und Wulf [PfWu97, Wulf97a] entwickelten beispielsweise die Konzepte *nutzungsbezogene Transparenz* und *Aushandelbarkeit* von Entscheidungen, die auch zum Konfliktmanagement in Vorgangsbearbeitungssystemen genutzt werden können [Rohd96].

Nutzungsbezogene Transparenz

Das Konzept der nutzungsbezogenen Transparenz bedeutet, dass ein Zugriff auf einem Objekt zwar erlaubt ist, jedoch bestimmte Benutzer, die Interesse an diesem Objekt bekundet haben, darüber informiert werden. Interessant sein könnte das z.B. beim Aktualisieren gemeinsam benutzter Dokumente wie gemeinsam benutzte Terminkalender. Aktualisiert „Benutzer A“ den Terminkalender, so würden unter Beachtung folgender Regel

`(Benutzer(„Benutzer A“) ∧ Dokument(„Terminkalender“) ∧ Operation(„ändern“), „erlaubt mit Transparenz“)`

bestimmte Benutzer benachrichtigt. Die betroffenen Benutzer müssen natürlich vorher ausgewählt werden (s.u.). Dies geschieht normalerweise beim Aufstellen bzw. Ändern einer Regel. Die Benachrichtigung kann synchron durch automatisches Öffnen eines Ereignisfensters auf den Bildschirmen der Betroffenen oder asynchron durch Versenden von Emails erfolgen.

Auf diese Weise werden Aktionen, die an Objekten von Interesse ausgeführt werden, transparent gemacht. Allerdings haben die betroffenen Benutzer keine Möglichkeit, die Aktion zu verhindern. Der in Bild 3 als Petri-Netz dargestellte Aktionsfluss verändert sich lediglich dahingehend, dass beim Ausführen der Aktion nicht nur der Aktivator, sondern auch die betroffenen Benutzer benachrichtigt werden.

Aushandlung

Aushandlungsmechanismen erweitern ein Zugriffskontrollsystem um die Möglichkeit der Abstimmung, ob eine Aktion ausgeführt werden darf oder nicht. Wie bei der Transparenz verwaltet auch hier das Zugriffskontrollsystem für jede Regel, deren Entscheidungswert „mit Aushandlung“ ist, eine Menge betroffener Benutzer, die beim Aufstellen der Regel manipuliert werden kann (siehe Abb. 4). Diese Benutzer werden im Falle eines gewünschten Zugriffs benachrichtigt. Die Benachrichtigung beinhaltet einen Stimmzettel (siehe Abb. 5), auf dem die betroffenen Benutzer wählen können, ob der Zugriff gestattet werden soll oder nicht. Das Zugriffskontrollsystem sammelt die abgegebenen Stimmzettel und entscheidet anschließend basierend auf dem Wahlausgang, ob der Zugriff gewährt werden soll und benachrichtigt den Benutzer, der die Aktion ausführen wollte. Die Aushandlung kann in unserer Implementation in Bezug auf eine prozentuale Zustimmungsquote konfiguriert werden.

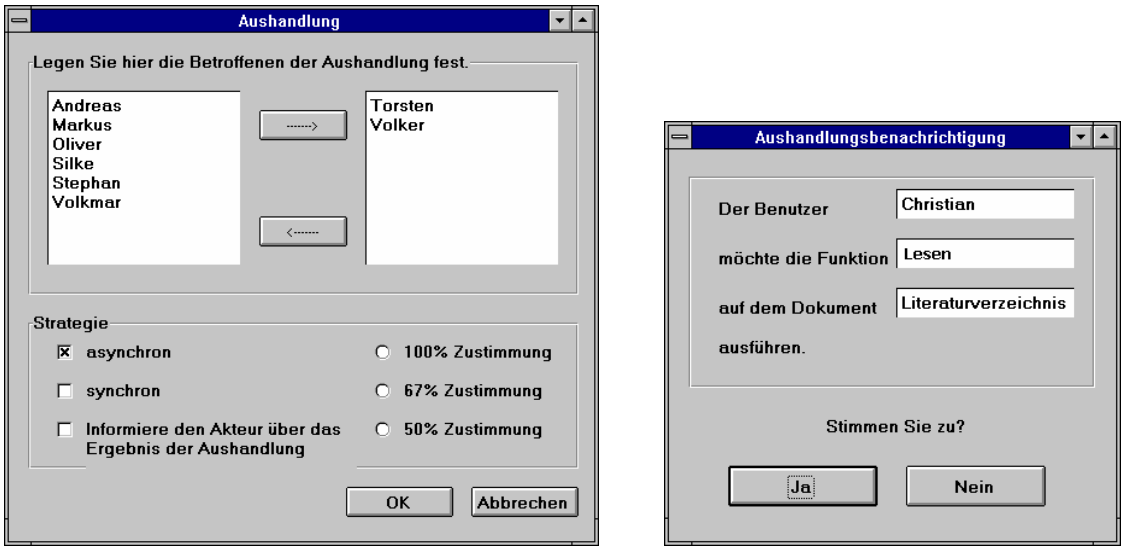

Bild 4, 5 Aushandlung

Durch diese Konzepte lässt sich die Mächtigkeit des Zugriffskontrollsystems und damit die Flexibilität des Gesamtsystems deutlich erweitern. In dem in Kapitel 5 beschriebenen Prototypen im Rahmen des POLITeam-Projekts [StWu98, Wulf97, Wulf98] wurden deshalb sowohl Transparenz- als auch Aushandlungsmechanismen integriert.

[6] In diesem speziellen Fall könnte man sich zur Vereinfachung mit der Verwendung von Gruppen behelfen. Diese müssten dann aber erst angelegt werden.

Repräsentation der Zugriffskontrollstrategie

Eine Zugriffskontrollstrategie wird im System dargestellt als eine Menge P von Zugriffsregeln. Bevor wir jedoch eine formale Definition geben, sollten noch einige wichtige Eigenschaften der Menge P diskutiert werden.

P sollte *vollständig* sein. D.h., dass in jeder möglichen Situation wenigstens eine Regel anwendbar sein muß, so dass das Zugriffskontrollsystem immer eine deterministische Entscheidung fällen kann, ob eine Operation erlaubt oder verboten ist. Dies kann auf einfache Weise dadurch erreicht werden, dass man der Menge P die triviale Regel (wahr, verboten) hinzufügt. Diese generellste aller Regeln kann immer angewendet werden. Sie besagt, dass alles, was nicht explizit erlaubt ist, verboten ist.

Außerdem soll die Auswertung von P deterministisch erfolgen, d.h., dass bei gleicher Zugriffskontrollstrategie und bei gleichen Situationen auch jedes Mal die gleiche Entscheidung getroffen wird. Dies ist eine Eigenschaft der Auswertungsstrategie.

Auswertungsstrategie

Wie oben beschrieben stellten die Interviewpartner Regeln und ganze Zugriffskontrollstrategien nach dem Prinzip „die speziellste Regel gilt“ auf. Deswegen soll in unserem Zugriffskontrollsystem diese Auswertungsstrategie angewendet werden, wobei darauf zu achten ist, dass der Determinismus nicht verletzt wird. Diese beiden Anforderungen sind nicht trivial in Einklang zu bringen, da es durchaus zwei Regeln geben kann, die beide in derselben Situation anwendbar sind und wobei nicht eine Regel die Ausnahme zur anderen darstellt. Wir bezeichnen solche Regeln als orthogonal zueinander. Im Folgenden stellen wir eine Auswertungsstrategie vor, die die obigen Anforderungen erfüllt.
Um deterministische Resultate zu erzielen, muss eine Auswertungsstrategie in jeder möglichen Situation alle anwendbaren Regeln prüfen und anschließend feststellen, welche dieser Regeln (genau eine) angewendet werden soll.

Der hier vorgestellte Ansatz basiert auf der eineindeutigen Gewichtung der Prädikate. Jedem Faktor ist ein Gewicht k zugeordnet, d h., eine wichtigeres Prädikat hat immer ein größeres Gewicht als ein unwichtigeres. Wendet man diese Zuordnung auf die in Tabelle 2 beschriebenen Faktoren an, so erhält man $Gewicht_{Dokumentname} = 8$ bis $Gewicht_{Operation} = 0$. Der Prioritätswert einer Regel wird dann nach folgender Formel ausgewertet:

$$Priorität_{(s,d)} := \sum_{\text{Prädikate } p \in s} 2^{\text{Gewichtung}_p}$$

Diese Formel liefert, wie leicht nachgewiesen werden kann, für Regeln, deren Gültigkeitsbereiche sich in mindestens einem Faktor unterscheiden, verschiedene Werte. Im folgenden Beispiel werden für zwei Regeln mit unterschiedlichen Bereichen die daraus resultierenden Prioritätswerte berechnet:

- as_1 = (Benutzer(„Benutzer A“) ∧ Dokumentname(„Text C“) ∧ Operation(„lesen“),erlaubt) → Priorität(as_1) = $2^5+2^8+2^0$ = 32+256+1 = 289
- as_2 = (Benutzerrolle(„Aushilfe") ∧ Dokumentname(„Text C“) ∧ Operation(„lesen“),verboten) → Priorität(as_2) = $2^4+2^8+2^0$ = 16+256+1 = 273

Es bleibt das Problem, mit Regeln umzugehen, deren Bereiche dieselben Prädikate aber unterschiedliche Argumente haben. Dies ist im folgenden Beispiel zu sehen:

- as_3 = (Benutzerrolle(„Buchhalter“) ∧ Dokumentname(„Text C“) ∧ Operation(„lesen“),erlaubt)
- as_4 = (Benutzerrolle(„Aushilfe“) ∧ Dokumentname(„Text C“) ∧ Operation(„lesen“),verboten)

In diesem Fall würde die oben vorgestellte Auswertungsstrategie für beide Regeln einen Prioritätswert von 273 ($= 2^8 + 2^4 + 2^0$) ermitteln. Würde nun ein Benutzer, der sowohl die Rolle „Buchhalter“ als auch die Rolle „Aushilfe“ innehat, auf Text C lesend zugreifen wollen, wären beide Regeln anwendbar und hätten auch den selben Prioritätswert. Dieses Problem, das unweigerlich zu nicht-deterministischen Ergebnissen führen würde, kann auf verschiedene Weise vermieden werden.

Zum einen könnte festgelegt werden, dass jeder Benutzer nur maximal eine Rolle einnehmen kann. Diese Lösung kann aber nur als sehr problematisch angesehen werden, da diverse Autoren (siehe auch [ShDe92]) auf die Wichtigkeit im CSCW-Bereich hinweisen, dass ein Benutzer mehrere Rollen einnehmen kann. In der hier vorgestellten Implementation wurde das Problem durch eine totale Ordnung aller Gruppen (Rollen, Organisationseinheiten etc.) gelöst. Die voreingestellte Ordnung bezieht sich hier auf das Erstellungsdatum einer Gruppe, d.h., ältere Gruppen haben eine höhere Priorität. Auf diese Weise kann der Determinismus erzwungen werden.

Für alle anderen Prädikate außer für sich überlappende Zeitintervallangaben gilt, dass verschiedene Argumente dazu führen, dass nur eine der Regeln mit gleichem Prioritätswert anwendbar ist.

- as_5 = (Benutzer („User A") ∧ Dokument(„Text C") ∧ Operation(„lesen"), erlaubt)
- as_6 = (Benutzer („User B") ∧ Dokument(„Text C") ∧ Operation(„lesen"), verboten)

Im Beispiel sieht man zwei gegensätzliche Regeln mit gleichem Prioritätswert. Allerdings kann nur maximal eine der beiden Regeln angewendet werden, da ein Benutzer nur einen Namen bzw. eine ID haben kann.

Zusammenfassend kann man sagen, dass P *gültig* ist genau dann, wenn für alle paarweise verschiedene Regeln (s_1,d_1) und (s_2,d_2) in P gilt, dass

- s_1 und s_2 sich durch mindestens ein Prädikat unterscheiden (→ diese beiden Regeln haben dann unterschiedliche Prioritätswerte) oder
- s_1 und s_2 mindestens ein Prädikat enthalten, dessen Argumente sich in den beiden Bereichen unterscheiden (beispielsweise enthalten beide Bereiche das Prädikat „Benutzerrolle", wobei im einen Bereich das Attribut „Mitarbeiter" im anderen das Attribut „Aushilfe" gesetzt ist) oder
- in s_1 und s_2 jeweils Zeitintervalle definiert werden, die sich nicht überlappen dürfen (bei gleichen Prädikaten mit gleichen Attributen und sich überlappenden Zeitangaben wären ansonsten mehrere Regeln mit gleichem Prioritätswert anwendbar).

Wenn P *gültig* ist, dann ist die oben beschriebene Auswertungsstrategie deterministisch. Auf Basis dessen kann nun der Begriff „Repräsentation einer Zugriffskontrollstrategie" formell definiert werden:

Eine Menge P von Zugriffsregeln ist die Repräsentation einer Zugriffskontrollstrategie genau dann, wenn P *gültig* und *vollständig* ist.

Definition 4 Repräsentation einer Zugriffskontrollstrategie

Metaregeln

In änderbaren Zugriffskontrollsystemen stellt sich die Frage, wer Zugriffskontrollstrategien aufstellen bzw. verändern darf. Normalerweise wird dieses Problem in Mehrbenutzersystemen so gehandhabt, dass nur der Eigentümer für seine Objekte Zugriffsrechte vergeben kann. Colouris und Dollimore [CoDo94] weisen jedoch darauf hin, dass dieser Ansatz für Gruppenarbeitssysteme zu unflexibel ist. Sie beschreiben die Notwendigkeit der Delegation der Rechtevergabe.

In dem hier vorgestellten Ansatz wird deswegen eine neue Operation (und das damit korrespondierende Recht) eingeführt: „manipulieren der Regeln eines Objekts". Dieses Recht benötigt ein Benutzer, um Regeln für ein Objekt aufstellen, löschen oder verändern zu können. Regeln, die über dieses neue Recht entscheiden, werden Metaregeln genannt. Für den Benutzer wird der Unterschied zwischen „normalen und Meta-Regeln nicht sichtbar, da sie in gleicher Weise dargestellt werden und auch manipuliert werden können.

Die klassische „Nur der Benutzer darf Rechte für ein Objekt vergeben"-Strategie sähe nun in der hier vorgestellten Zugriffskontrolle folgendermaßen aus:

- (Operation(„manipulieren der Regeln eines Objekts"), verboten)
- (Benutzer-Dokument-Beziehung(„Eigentümer") ∧ Operation („manipulieren der Regeln eines Objekts"), erlaubt)

Weiterhin führen wir für sogenannte Containerobjekte (Verzeichnis, Ordner etc.) eine weitere Operation „manipulieren der Regeln für den Inhalt" ein. Mithilfe dieser Operation bzw. dem dazugehörigen Recht lassen sich Zugriffskontrollstrategien für Hierarchien und auf diese Weise gruppierte Objekte erzeugen. Beispielsweise besagen folgende Regeln, dass die Mitarbeiter der Verwaltung für Objekte im Ordner „Gemeinsam benutzte Dokumente" Regeln aufstellen dürfen:

- (Benutzerrolle(„Mitarbeiter der Verwaltung") ∧ Operation („manipulieren der Regeln für den Inhalt") ∧ Dokument(„Gemeinsam benutzte Dateien",erlaubt)

Die hier beschriebene Lösung, dass Metaregeln nur dazu dienen festzulegen, wer die Zugriffskontrollstrategie in Bezug auf ein ausgewähltes Objekt verändern darf, ist nur ein sehr einfaches Beispiel, wie eine discretionary access control eingeschränkt werden kann. Metaregeln können auch – falls man sie feingranularer formuliert – dazu dienen, die Veränderbarkeit bestimmter Regeln zu limitieren. Auf diese Weise können beispielsweise organisationsinterne Normen bez. der Zugriffskontrollstrategie als Regeln definiert sein. Eine Metaregel besagt in diesem Fall, dass bestimmte Regeln nicht änderbar sind. Benutzer können dann weitere Regeln, die die Zugriffskontrollstrategie verändern, anlegen. Weiterhin muss dafür gesorgt sein, dass die solche organisationsweit geltenden Regeln von der Priorität immer höher eingestuft werden als die von den Benutzern formulierten. So liesse sich eine discretionary access control im Rahmen einer mandatory access control implementieren. Dieser Ansatz soll jedoch in diesem Aufsatz nicht weiter verfolgt werden.

5. Beschreibung der Implementation

Das oben beschriebene Zugriffskontrollsystem wurde im Rahmen des POLITeam-Projektes innerhalb des kommerziellen Gruppenarbeitssystems LINKWORKS implementiert.

Eines der Probleme bei der Implementation war die Repräsentation der Einzelfaktoren, die den Bereich einer Regel bestimmen. Einige dieser Faktoren wie der Benutzername sind unter LINKWORKS standardmäßig verfügbar, andere, speziell Gruppen, Rollen etc., nicht. Deswegen wurde innerhalb von LINKWORKS eine neue Objektklasse erzeugt: die Gruppenklasse. Objekte dieser Klasse bestehen aus einem eindeutigen Namen und einer Liste von Mitgliedern (Benutzern). Diese Gruppenobjekte können innerhalb einer Regel als Attribut für das Prädikat „Benutzer“ eingesetzt werden. Jeder Benutzer kann Gruppenobjekte erstellen, verändern oder löschen, falls es ihm gemäß der Zugriffsrechte erlaubt ist. Auf diese Weise können Gruppen verwendet werden, um diverse Konzepte wie Rollen (einelementige Gruppen), Vertrauensbeziehungen („Freunde“) oder auch Organisationseinheiten zu definieren.

Dokumente können auf zweierlei Weise gruppiert werden: Zum einen ist es möglich, im Bereich einer Regel eine ganze Objektklasse anzugeben. Hier würde also die Gruppierung über die Klassifizierung vorgenommen. Eine andere Möglichkeit bieten die Containerobjekte unter LINKWORKS. Liegen verschiedenartige Dokumente im gleichen Containerobjekt, so können für sie als Gruppe Regeln aufgestellt werden, indem man Regeln für den Inhalt des Containerobjekts definiert.

Präsentation der Zugriffskontrollstrategie

Das Zugriffskontrollsystem wurde in LINKWORKS so integriert, dass man sich über Menüfunktionen sowohl Regeln für ein Objekt als auch für den Inhalt eines Containers anzeigen lassen kann. Beide Funktionen werden mit derselben Benutzerschnittstelle dargestellt. Bild 6 zeigt die Benutzerschnittstelle. Es werden die aktuellen Regeln, die „Text C“ betreffen, in natürlicher Sprache und sortiert nach Prioritätswerten angezeigt.

Bild 6 Präsentation der Zugriffsregeln

Mithilfe des Buttons „Zugriffsverhalten ausprobieren“ gelangt man in den Explorationsmodus. Hier kann man prüfen, inwieweit das gewollte Zugriffsprofil mit dem tatsächlichen übereinstimmt, indem man hypothetische Situationen künstlich erzeugt (Was passiert, wenn Benutzer D lesend auf Text C zugreifen will?).

Die Manipulations-Benutzerschnittstelle

Bild 7 Manipulation einer Zugriffsregel

Die oben beschriebene Benutzerschnittstelle stellt weiterhin Möglichkeiten zur Verfügung, mit denen sich Zugriffskontrollstrategien manipulieren lassen („Regel löschen“, „Neue Regel“, „Regel ändern“). Bild 7 zeigt die Benutzerschnittstelle zum Ändern oder Erstellen von Regeln. Der Benutzer kann hier den Bereich und auch die Entscheidung auf recht einfache Weise (Auswahlboxen) zusammenstellen. Hilfreich ist dabei vor allem, dass die aktuell konstruierte Regel im unteren Teil des Fensters in Textform angezeigt wird. Diese Funktion erlaubt es auch unerfahrenen Benutzern, Fehler selbst zu erkennen und zu korrigieren [Stie96].

6. Evaluation

Die Vorgehensweise in der abschließenden Evaluation des Prototypen orientierte sich an der „Thinking Aloud“-Methode [Niel93]. In erster Linie ging es bei dem Test darum, Ergebnisse bez. der Benutzerschnittstelle und des zugrunde liegenden Zugriffskontrollsystems zu gewinnen. Die grundlegende Idee der Methodik ist, die Testpersonen Aufgaben mit dem Prototypen bearbeiten zu lassen, wobei sie über ihre Interpretation der Präsentation und der Manipulationsmöglichkeiten laut „nachdenken“ sollen. Diese Methode erlaubt es, auf einfache und effiziente Weise Probleme bei der Handhabbarkeit aufzudecken.

Die Tests fanden unter Laborbedingungen statt. Die Interviews wurden mittels Tonbandgerät aufgezeichnet. Unter Berücksichtigung zusätzlicher während der Interviews gemachter Notizen wurden die Tonbandaufnahmen anschließend transkribiert und inhaltsanalytisch ausgewertet.

Die Benutzer deckten verschiedene Kompetenzstufen ab (Entwickler, Power-User und Einsteiger). Anhand eines kleinen Testszenarios, ein Textdokument „Tagebuch“ und eine Gruppe „Freunde“, sollten die Testpersonen die Effektivität der Präsentation und der Manipulation beurteilen. Die erste Aufgabe war die Beschreibung einer schon vorhandenen Zugriffskontrollstrategie in eigenen Worten. Anschließend wurden den Benutzern verschiedene Fragen wie „Darf ‚Benutzer C‘, der zur Gruppe ‚Freunde‘ gehört, auf das Tagebuch lesend zugreifen?“ gestellt, die sie unter Beachtung der aktuellen Zugriffskontrollstrategien beantworten sollten. Im zweiten Teil bestand die Aufgabe darin, verschiedene Zugriffskontrollstrategien selbst einzurichten. Eine detailliertere Beschreibung der Tests findet sich in [Stie96, WSP99].

Die Ergebnisse der Evaluation bestätigten den hier beschriebenen Ansatz, obwohl die Gebrauchstauglichkeit der Benutzerschnittstellen noch nicht vollkommen zufriedenstellend war. Speziell die Ansätze, ein regelbasiertes Zugriffskontrollsystem zu verwenden und Zugriffkontrollstrategien in natürlicher Sprache zu formulieren, wurden von den Testpersonen verstanden.

Im ersten Teil der Tests (Präsentation) hatten einige Benutzer Schwierigkeiten, die vorhandenen Zugriffskontrollstrategien zu beschreiben. Einer der Benutzer verstand die Ordnung der Regeln nach Generalität (Einschränkung des Bereichs) nicht. Trotzdem beschrieb er die Zugriffskontrollstrategie richtig, obwohl bei ihm die Auswertung deutlich länger dauerte. Sowohl die befragten Power-User als auch die Entwickler hatten jedoch kaum Probleme, die Zugriffskontrollstrategie zu beschreiben. Um die Verständlichkeit noch zu erhöhen, sollte eventuell die Generalität der Regeln deutlicher in der Benutzerschnittstelle kenntlich gemacht werden. Ein anderer Ansatz wäre, den Explorationsmodus zu erweitern, um gerade unerfahreneren Benutzern den Einstieg zu erleichtern.

Der zweite Teil der Tests verlief etwas überraschend, da die Benutzer sehr unterschiedliche Wege fanden, um dieselben Zugriffskontrollstrategien zu erzeugen. Das Ergebnis war dann, dass die Benutzer weniger Probleme damit hatten, bestimmte Zugriffskontrollstrategien zu erstellen als sich zwischen verschiedenen möglichen Alternativen zu entscheiden.

7. Zusammenfassung und Ausblick

In diesem Aufsatz wurde ein Zugriffskontrollsystem für asynchrone dokumentenorientierte Gruppenarbeitssysteme vorgestellt. Die Hauptanforderungen bei der Entwicklung des Modells waren, dass sowohl die Präsentation als auch die Manipulationen der Strategien leicht nachvollziehbar und durchführbar sind. Dies ist deswegen besonders wichtig, da gerade Systeme, die Kooperation der Teilnehmer unterstützen sollen, sehr flexibel hinsichtlich der Zugriffsrechte sein müssen. Das Design basierte auf Ergebnissen einer Feldstudie zu diesem Thema. Weiterhin wurden in unserem Modell Anforderungen, die auf den Forschungsergebnissen anderer Arbeiten basieren (Negative Rechte, Benutzerrollen und Delegation), berücksichtigt. Den Abschluß bildet eine Diskussion und Bewertung der Ergebnisse einer Evaluation, die sich mit der Gestaltung der Benutzerschnittstelle und auch dem zugrundeliegenden Konzept befasste.

Die Evaluation fand unter Laborbedingungen statt. Im Vordergrund stand dabei die Untersuchung der Effektivität der Präsentation und Manipulation. Dabei blieb die Frage offen, inwieweit tatsächlich alle in den Anwendungsfeldern gewünschten Zugriffskontrollstrategien ausgedrückt werden können. Um dies zu untersuchen, sind weitere Feldtests nötig. Ein Ergebnis einer solchen Studie könnte beispielsweise sein, dass dem hier vorgestellten Zugriffskontrollsystem weitere den Bereich bestimmende Faktoren hinzugefügt werden müssen oder die Gewichtung der einzelnen Faktoren je nach Anwendungsfeld geändert werden muss.

8. Danksagung

Dank gebührt Prof. Armin B. Cremers, Torsten Engelskirchen, Helge Kahler, Matthias Krings, Andreas Pfeifer, Volkmar Pipek, Markus Rittenbruch und Markus Rohde für die anregende Diskussion des hier präsentierten Ansatzes.

9. Bibliographie

[CoDo94] Coulouris, G.; Dollimore, J.: Requirements for security in cooperative work: two case studies. Technical Report 671, Department of Computer Science, Queen Mary and Westfield College, University of London, 1994 (http://www.dcs.qmw.ac.uk /research/distrib/).

[CoDo95] Coulouris, G.; Dollimore, J.: Protection of shared objects for cooperative work. Technical Report 703, Department of Computer Science, Queen Mary and Westfield College, University of London, 1995 (http://www.dcs.qmw.ac.uk/research/distrib/).

[Crem98] Cremers, A. B., Kahler, H., Pfeifer, A., Stiemerling, O., and Wulf, V., "PoliTeam - Kokonstruktive und evolutionäre Entwicklung einer Groupware," *Informatik Spektrum*, vol. 21, S. 194-202, 1998.

[Rann97] Rannenberg, K., Pfitzmann, A., and Müller, G., "Sicherheit, insbesondere mehrseitige IT-Sicherheit," in *Mehrseitige Sicherheit in der Kommunikationstechnik*, G. Pfitzmann and A. Müller, Eds. Bonn: Addison-Wesley-Longman, S. 21-29, 1997.

[VoKe83] Voydock, V. L. and Kent, S. T., "Security Mechanisms in High-Level Network Protocols," *ACM Computing Surveys*, vol. 15, No. 2, S. 135-170, 1983.

[DEC95] Digital Equipment: LINKWORKS 3.0 Administration Guide, DEC, 1995.

[DeSh98] Dewan, P., Shen, H.: Flexible Meta Access-Control for Collaborative Applications, Proceedings of CSCW'98, ACM Press, New York, S. 247-256, 1998.

[Elli91] Ellis, C.; Gibbs, S.J.; Rein, G.L.: Groupware: Some Issues, Some Experiences. Communications of the ACM, Vol. 44, No. 1, S. 39-58, 1991.

[FSW81] Fernandez, E.B.; Summers, R.C.; Wood, C.: Database Security and Integrity, Addison Wesley, S. 65 ff, 1981.

[GrSa86] Greif, I.; Sarin, S.: Data Sharing in Group Work. Proceedings of CSCW '86, ACM Press, New York, S. 175-183, 1986.

[Kloe95] Klöckner, K.; Mambrey, P.; Sohlenkamp, M.; Prinz, W.; Fuchs, L.; Kolvenbach, S.; Pankoke-Babatz, U.; Syri, A.: POLITeam - Bridging the Gap between Bonn and Berlin for and with the Users. Proceedings of ECSCW '95, Kluwer , S. 17-32, 1995.

[Lamp74] Lampson, B.W.: Protection. ACM Operating System Review, Vol. 8, S. 18-24, 1974.

[Micr97| Microsoft Corporation: Microsoft Windows NT Server Version 4.0 – Die technische Referenz, Microsoft Press, ISBN: 3-86063-241-8, 1997.

[Mins93] Minsky, N.H.: A Law-Governed Protection Mechanism for Distributed Systems. Technical Report, Rutgers University, LCSR, 1996 (http://www.cs.rutgers.edu/ ~minsky/public-papers/).

[Neme97] Nemeth, E.: Unix Systemverwaltung. Prentice-Hall, ISBN: 3-8272-9511-4, 1997.

[Niel93] Nielsen, J.: *Usability Engineering*. AP Professional, New York, 1993.

[PiWu99] Pipek, V.; Wulf, V.: POLITeam - Konzepte zur Einführung von Groupware, in: Kubicek, H. u.a. (Hrsg.): Multimedia@Verwaltung – Jahrbuch, Telekommunikation und Gesellschaft, Hüthig, Heidelberg 1999, pp. 389 – 390.

[PfWu97] Pfeifer, A.; Wulf, V.: Negotiating Conflicts in Active Databases. In: Proceedings of the 4th International Conference on Concurrent Engineering (CE ´97), 20.-22.8.97, Oakland University, Rochester, MI., Technomic Publishing, Lancaster, 1997.

[Rohd96] Rohde M., Pfeifer A., Wulf V.: Konfliktmanagement bei Vorgangsbearbeitungssystemen. In: Wirtschafts-informatik, No. 2, 1996, S. 199-208.

[Saty90] Satyanaranayan, M.: Scalable, secure and highly available distributed file access. IEEE Computer, 23(5), S. 9-22, 1990.

[ShDe92] Shen, H.; Dewan, P.: *Access Control for Collaborative Environments*. In Computer-Supported Cooperated Work '92, Sharing Perspectives. Proceedings of the Conference on Computer Supported Cooperative Work, (ACM Press, New York), S. 51-58, 1992.

[Sikk97] Sikkel, K.: A Group-based Authorization Model for Cooperative Systems. In Proceedings of ECSCW '97, Lancaster, Kluwer, S. 345-360, 1997.

[SiSt98] Sikkel, K.; Stiemerling, O.: User-Oriented Authorization in Collaborative Environments. In: Proceedings of COOP '98, 26.-29.5.98, Cannes, France, S. 175-183, 1998.

[Stie96] Stiemerling, O.: *Anpaßbarkeit in Groupware - ein regelbasierter Ansatz*. Diplomarbeit, Institut für Informatik III, Universität Bonn, http://www.informatik.uni-bonn.de/~os/Publications/MasterThesis.ps , 1996.

[Stie97] Stiemerling, O.; Kahler, H. and Wulf, V.: How to Make Software Softer - Designing Tailorable Applications. In Proceedings of 2nd Conference on the Design of Interactive Systems, Amsterdam (NL), ACM Press, S. 365-376, 1997.

[StWu98] Stiemerling, O.; Wulf, V.: Beyond 'Yes or No' - Extending Access Control in Groupware with Negotiation and Awareness, In: Proceedings of COOP '98, 26.-29.5.98, Cannes, France, S. 111-120, 1998.

[WSP99] Wulf, V .; Stiemerling, O.; Pfeifer, A.: Tailoring Groupware for Different Scopes of Validity. In: Behaviour & Information, 1999.

[Wulf74] Wulf, W.; Cohen, E.; Corwin, W.; Jones, A.; Levin, R.; Pierson, C.; Pollack, F.: HYDRA: The kernel of a multiprocessor operating system. Communication of the ACM, 17(6), S. 337-345, 1974.

[Wulf97] Wulf, V.: Organisatorischer Wandel bei Einführung von Groupware. In: Proceedings der dritten internationalen Tagung "Wirtschaftsinformatik ´97" am 26.2. - 28.2.97,Berlin, S. 167 – 182, 1997.

[Wulf97a] Wulf, Volker: Konfliktmanagement bei Groupware, Vieweg, Braunschweig und Wiesbaden 1997.

[Wulf98] Wulf, V.: *On Conflicts and Negotiation in Multiuser Application*. In: Kent, A.; Williams, J. G. (eds): Encyclopedia of Microcomputers, Dekker, New York, 1999.