Ein Zugriffskontrollmodell für aufgabenbezogene Rollen

Dissertation

zur Erlangung des akademischen Grades

Dr. rer. pol.

vorgelegt an der

Fakultät für Wirtschaftsinformatik und Angewandte Informatik

der Otto-Friedrich-Universität Bamberg

von

Gerlinde Fischer

Bamberg, November 2015

Gutachter:

Prof. Dr. Elmar J. Sinz

Prof. Dr. Sven Overhage

Disputation: 10. Juli 2015

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Definitionsverzeichnis

Abkürzungsverzeichnis

A	Aufgabe
A_p	Nicht automatisierbare Aufgabe
A_m	Automatisierbare Aufgabe
AT_p	Personeller Aufgabenträger
AT_m	Maschineller Aufgabenträger
ACI	Access Control Information
ABAC	Attributbasierte Zugriffskontrolle
ACL	Access Control List
ACM	Access Control Matrix
ADF	Access Decision Function
ADK	Anwendungsfunktionen – Datenverwaltung – Kommunikation
AEF	Access Enforcement Function
ANSI	American National Standards Institute
ARBAC	Administration im rollenbasierten Zugriffskontrollmodell
ARH	Administrationsrollenhierarchie
ARo	Administrationsrolle
ASZ	Administrationssubjektzuordnung
AwS	Anwendungssystem
AZ	Administrationszugriffsrecht
AZZ	Administrationszugriffsrechtszuordnung
BSI	Bundesamt für Sicherheit in der Informationstechnik
BDSG	Bundesdatenschutzgesetz
Bit	Binary Digit
C	Zertifizierung (Certification)
C-C	Computer-Computer Kommunikation
CDI	Constraint Data Item
CSS	Cascading Style Sheet
DAT	Dynamische Aufgabentrennung
DBMS	Datenbankmanagementsystem
DNA	Desoxyribonukleinsäure engl.: **D**esoxyribo**n**ucleic **a**cid
DV	Datenverarbeitung

E	Durchführungsregeln (Enforcement)
EER	Gleichfehlerrate
eRBAC	Erweitertes rollenbasiertes Zugriffskontrollmodell
FAR	Falschaktzeptanzrate
FFR	Falschrückweisungsrate
FN2	FlexNow 2
FN2AUTH	Authentifizierungsportal FlexNow
FN2LM	Dozentenportal (Lehrstuhlmodul) FlexNow
FN2RBAC	Rollenbasierte Zugriffskontrolle in FlexNow 2
FN2STUD	Studierendenschnittstelle FlexNow
FN2XML	XML-Server von FN2
HRU	Harrison, Ruzzo, Ullman Modell
IEC	International Electrotechnical Commission
INCITS	International Committee for Information Technology Standards
IS	Betriebliches Informationssystem
ISO	International Organisation of Standardization
IVP	Integrity Verification Procedure
IT	Informationstechnologie
LDAP	Lightweight Directory Access Protocol
LM	Lehrstuhlmodul
M-C	Mensch-Computer-Kommunikation
MDStV	Mediendienste-Staatsvertrag
M-M	Mensch-Mensch-Kommunikation
NCSC	National Cyber Security Center
NIST	National Institute of Standards and Technology
O	Objekt
Op	Operator
OT	Objekttyp
ORM	Organisations- und Ressourcenmanagement
P	Parameter
PA	Prüfungsamt
PAVOR	Prüfungsausschussvorsitzender

PBDM	Permission-based Delegationsmodel
PDF	Portable Document Format
PIN	Persönliche Identifikationsnummer
PO	Prüfungsordnung
PD	Prüfungsdurchführung
RBAC	Rolebased Access Control
RBDM	Rollenbasiertes Delegationsmodell
RDM	Rollenbasiertes Delegationsmodell
RFID	Radio Frequency Identification
RH	Rollenhierarchie
Ro	Rolle
ROB	Rollenbereich
RT	Rollentyp
S	Subjekt
SAT	Statische Aufgabentrennung
Si	Sitzung
SK	Sicherheitsklasse
SOM	Semantisches Objektmodell
Sw	Schlüsselwert
SZ	Subjektzuordnung
TAN	Transaktionsnummer
TDDSG	Teledienstedatenschutzgesetz
TKG	Telekommunikationsgesetz
TP	Transformation Procedure
UDI	Unconstraint Data Item
VB	Vorbedingung
WfMC	Workflow Management Coalition
SAT	Statische Aufgabentrennung
Z	Zugriffsrecht
ZK	Zugriffsklasse
ZZ	Zugriffsrechtszuordnung

1 Einleitung

Die zunehmende Vernetzung und Integration von Informationssystemen in Wirtschaft und Verwaltung führt immer wieder zu der Frage, wie die Sicherheit der Informationen in Anwendungssystemen gewährleistet werden kann. Ausgangspunkt dieser Arbeit bildet in diesem Kontext die Entwicklung eines Zugriffskontrollsystems für Anwendungssysteme (AwS). Dazu muss als Grundlage für das Zugriffskontrollsystem ein Zugriffskontrollmodell gefunden werden, das für den Einsatz im betrieblichen Informationssystem (IS) geeignet ist.

Im IS werden Informationen in den Organisationen von Wirtschaft und Verwaltung sowie auch überbetrieblich verarbeitet, z. B: erfasst, gespeichert, transformiert und bereitgestellt (Ferstl und Sinz 2013, S. 3). „Betriebliche Informationssysteme dienen“ dabei „der Lenkung betrieblicher Prozesse oder erstellen Dienstleistungen in Form von Informationen“ (Ferstl und Sinz 2013, S. 12). Das IS ist ein in der Wirtschaftsinformatik unter verschiedensten Aspekten gut untersuchtes Forschungsgebiet[1]. Ein Beispiel für eine untersuchte Dimension ist die Aufgaben- und Aufgabenträgerebene des betrieblichen Informationssystems (Ferstl und Sinz 2013, S. 3).

Die Untersuchung der Zugriffskontrolle zum Schutz von Informationen vor unberechtigten Zugriffen ist ein Teilgebiet der Sicherheit in Informationssystemen. Die Zugriffskontrolle wird oftmals mit Hilfe von formalen Modellen, den Zugriffskontrollmodellen, beschrieben. Diese sind bereits seit Jahrzehnten Gegenstand der Forschung. Eines der ersten auch formal untersuchten Zugriffskontrollmodelle ist die Zugriffsmatrix (Graham und Denning 1971). Unter den zuletzt entwickelten Zugriffskontrollmodellen befindet sich das rollenbasierte Zugriffskontrollmodell (RBAC) (Ferraiolo und Kuhn 1992; ANSI INCITS 359-2004 2004). Dieses knüpft die Erlaubnis eines Zugriffs nicht direkt an Personen, sondern an Rollen. Die Vergabe von Berechtigungen mit Hilfe von Rollen stellt dabei die durchzuführenden Aufgaben in den Mittelpunkt (Eckert 2012, S. 272f).

Oft befassen sich die Forschungen im Bereich der Zugriffskontrollmodelle ausschließlich mit technischen Aspekten, d. h. wie mit einem Zugriffskontrollmodell

[1] Eine ausgewählte Aufzählung findet sich in Ferstl und Sinz (2013, S. 11f).

eine sichere Zugriffskontrolle modelliert und in einem Zugriffskontrollsystem umgesetzt werden kann. Der Prozess zum Auffinden der Rollen und Zugriffsrechte anhand von Aufgaben wird meist nur unzureichend beschrieben. Die Untersuchungen im IS, wie die Zuordnung der Aufgaben zu Geschäftsprozessen sowie die Strukturierung von Aufgaben und Aufgabenträgern, können einen Beitrag leisten die Definition der Rollen methodisch zu unterstützen.

Das Konzept der Rolle in das Konzept des betrieblichen Informationssystems zu integrieren wurde bislang in der Forschung weitgehend vernachlässigt. Die vorliegende Arbeit schließt diese Lücke durch Verbindung der beiden Forschungsgebiete, indem vor allem die Beziehung zwischen Rolle, Aufgabe und Aufgabenträger methodisch untersucht wird.

1.1 Problemstellung und Motivation

Bei der Auswahl eines geeigneten Zugriffskontrollmodells und der Entwicklung eines Zugriffskontrollsystems für Anwendungssysteme treten folgende Problemstellungen auf:

Permanent notwendige Weiterentwicklung der Informationssicherheit

Durch sich ständig ändernde Nutzungsszenarien und technologische Weiterentwicklungen, wie z. B. die Verbreitung des Internets, ein hoher Grad an Vernetzung, der Nutzung von Webservices oder Cloud Computing (Feng et al. 2004, S. 357; Linkies und Off 2006; Strembeck und Neumann 2004, S. 393; Eymann 2013) ergibt sich immer wieder neuer Forschungsbedarf in Bezug auf Informationssicherheit und Zugriffskontrolle.

Heterogene Authentifizierungs- und Zugriffskontrollsysteme

Durch den Einsatz verschiedener Anwendungssysteme existieren in Unternehmen und Verwaltungen oft für jedes Anwendungssystem autonome Authentifizierungs- und/oder Zugriffskontrollsysteme. Weiterhin gibt es keinen allgemein akzeptierten Standard für die Ermittlung, Speicherung und Überprüfung von Berechtigungsstrukturen. Deshalb haben sich in Unternehmen eine Vielfalt von Lösungsansätzen entwickelt (Linkies und Off 2006, S. 22). Diese Koexistenz von Zugriffskontrollsystemen kann zu inkonsistenter und redundanter Datenhaltung von Nutzer-, Zugangsdaten und Zugriffskontrollinformationen führen und er-

zeugt durch die notwendige Synchronisation der entsprechenden Daten einen hohen administrativen Aufwand (Herwig und Schlabitz 2004, S. 290).

Attribute aus dem Anwendungssystem im Zugriffskontrollsystem

Notwendige Voraussetzung für die Zugriffskontrolle ist eine sichere Authentifizierung. Aufgrund der Bemühungen, die Authentifizierung in Organisationen als Single-Sign-On-Verfahren durchzuführen, ist zwar die Identität des Nutzers bekannt, allerdings stehen Attribute aus dem AwS nach der Authentifizierung nicht automatisch zur Verfügung. Eben diese sind jedoch erforderlich, damit u. a. Informationen bei der Aufgabendurchführung auf den erlaubten Wertebereich eingeschränkt werden können.

Bei der Untersuchung der Rolle zur Integration in das IS sind folgende Problemstellungen zu finden:

Homonymkonflikt in Bezug auf den Begriff Rolle

Der Begriff Rolle wird in der Soziologie, der Organisationslehre, den Sozialwissenschaften sowie der Informatik verwendet. Es existiert selbst in der Informatik keine Übereinstimmung über die verschiedenen Bedeutungen des Begriffs Rolle, die alle Nutzungen einschließt (Boella et al. 2007, S. 81; Neumann und Strembeck 2001, S. 58). Daraus resultiert ein Homonymkonflikt in Bezug auf den Begriff Rolle.

Mangelnde Untersuchung der Rolle im IS

Zu den folgenden exemplarisch in der Literatur zu findenden Aussagen fehlen oftmals methodisch fundierte Untersuchungen:

- „Bei einer rollenbasierten Modellierung werden die Berechtigungen zur Nutzung geschützter Komponenten direkt an Rollen und damit an Aufgaben geknüpft. Die durch Rollen modellierten Aufgaben werden von Subjekten durchgeführt, so dass das Sicherheitsmodell festlegen muss, welche Subjekte welche Aufgaben durchführen, d. h. in welchen Rollen agieren“ (Eckert 2012, S. 272f).
- Um einen umfassenden Schutz der Informationen zu gewährleisten ist es erforderlich, dass sich die Zugriffsrechte auf die in einem Anwendungssystem abgebildeten Aufgaben beziehen (Beresnevichiene 2003, S. 12).

- „.., roles are used to model different job positions and scopes of duty … within an information system“ (Prescher et al. 2014).

1.2 Untersuchungsgegenstand, Zielsetzung und Lösungsansatz

Untersuchungsgegenstand der vorliegenden Arbeit ist die IT-Sicherheit, ein Teilbereich der Informationssicherheit im betrieblichen IS. Das Hauptaugenmerk liegt auf teilautomatisierten Aufgaben und damit der Schnittstelle zwischen personellen und maschinellen Aufgabenträgern. Ein personeller Aufgabenträger benötigt zur Erledigung einer teilautomatisierten Aufgabe Informationen, die in einem Anwendungssystem verwaltet werden. Dabei muss gewährleistet sein, dass einem personellen Aufgabenträger alle notwendigen Informationen für die Erledigung seiner Aufgaben zur Verfügung stehen und er ausschließlich Informationen erhält, für die eine Autorisierung vorliegt.

Beispielsweise möchte ein Prüfer[2] Noten erfassen. Dazu benötigt er die Liste der Prüfungsteilnehmer. In ihr dürfen nur Studierende angezeigt werden, die bei diesem Prüfer eine Prüfung abgelegt haben. In Abb.1-1 wird dieses Szenario dargestellt:

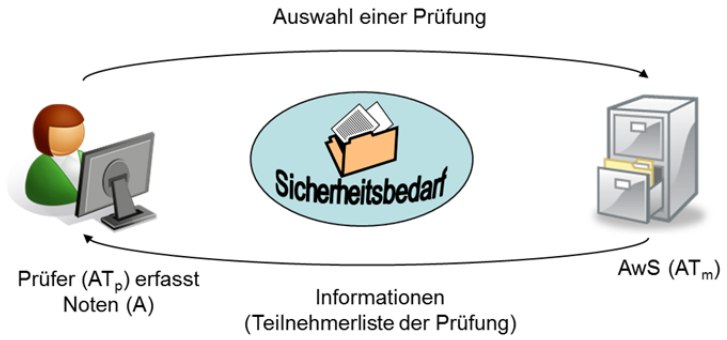

Abb.1-1 Aufgaben (A), personeller Aufgabenträger (AT_p), maschineller Aufgabenträger (AT_m) und Informationen im betrieblichen Informationssystem

Zielsetzung der vorliegenden Arbeit ist eine systematische Integration des Konzeptes der Rolle in das Konzept des betrieblichen Informationssystems. Ausgangspunkt dafür ist die Untersuchung der Beziehung zwischen Aufgaben, Aufgabenträgern und Rolle. Es soll gezeigt werden, dass Rollen in einem Top-Down-Ansatz aus der Aufgabenebene des IS systematisch entwickelbar sind. Zusätzlich soll mit diesen Erkenntnissen ein erweitertes Zugriffskontrollmodell auf Basis von RBAC entwickelt

[2] Die gewählte Form des Geschlechts bezieht immer auch das jeweils andere Geschlecht in vollem Umfang mit ein.

werden, um ein Zugriffskontrollsystem für Anwendungssysteme implementieren zu können.

Nachstehende Restriktionen sind von einem Zugriffskontrollmodell für aufgabenbezogene Rollen zur Zugriffskontrolle für Anwendungssysteme zu erfüllen:

- Das Zugriffskontrollmodell muss die Sicherheitsziele eines Unternehmens umsetzen.
- Zusätzlich muss es mit Attributen des Nutzers aus dem Anwendungssystem parametrisierbar sein, damit eine Personalisierung vorgenommen und die Domäne der Informationen bei Bedarf beschränkt werden kann.
- Die Konstruktionsprinzipien sicherer Informationssysteme sollen berücksichtigt werden.
- Das Zugriffskontrollsystem auf Basis eines Zugriffskontrollmodells sollte in heterogenen Anwendungslandschaften einsetzbar sein.

Als Einstieg wird eine Terminologie der Informationssicherheit im Allgemeinen, der Authentifizierung, Kryptologie und Zugriffskontrolle einschließlich Zugriffskontrollmodellen und Zugriffskontrollsystemen im Besonderen definiert. Daneben werden Konstruktionsprinzipien für sichere Zugriffskontrollsysteme vorgestellt. Um ein geeignetes Zugriffskontrollmodell für den Einsatz in einem Zugriffskontrollsystem für Anwendungssysteme auszuwählen, wird ein Klassifikationsrahmen erstellt. In diesen werden die untersuchten Zugriffskontrollmodelle eingeordnet und abschließend bewertet. Für die Entwicklung eines Zugriffskontrollsystems wurde als Grundlage das Referenzmodell des rollenbasierten Zugriffskontrollmodells (RBAC) ausgewählt. Da der Bedarf einer Erweiterung des Referenzmodells identifiziert wurde, werden im nächsten Schritt bestehende erweiternde Konzepte für RBAC näher untersucht.

Der zentrale Bereich dieser Arbeit verbindet das Konzept der Rolle mit dem Konzept des betrieblichen Informationssystems und untersucht zunächst den Homonymkonflikt des Begriffs Rolle, um ihn durch Einordnung in Rollenkonzepte zu strukturieren. Aus den einzelnen Konzepten wird ein ganzheitliches Rollenkonzept abgeleitet und untersucht, ob es sich um Anwendungssysteme und Funktionen ergänzen lässt. Nach der Beantwortung der Frage, ob die Rolle vom Typ Aufgabe oder Aufgabenträger ist, wird das Virtualisierungskonzept auf das Konzept der Rolle übertragen. Im

nächsten Schritt werden Aufgaben, Aufgabenträger und Rollen in die Informationsbeziehungen und Kommunikationssysteme des IS eingeordnet.

Durch die abschließende Einordnung in die Unternehmensarchitektur nach der SOM-Methodik[3] und der Erweiterung des Metamodells von der Zuordnung von Aufgaben zu Aufgabenträgern um die Rolle soll überprüft werden, ob Rollen aus den Aufgaben und den organisatorischen Rahmenbedingungen in Unternehmen im Top-Down-Ansatz heraus entwickelt werden können.

Als Grundlage für das zu entwickelnde Zugriffskontrollsystem wird RBAC zu einem erweiterten Zugriffskontrollmodell (eRBAC) vervollständigt, um flexible Aufrufstrukturen und Personalisierung mit RBAC in einem Zugriffskontrollsystem zu ermöglichen. Das Umfeld, in dem diese Arbeit entstanden ist, bildet die Domäne der Prüfungsverwaltung. Als Proof of Concept wird das prototypisch realisierte Zugriffskontrollsystem für das Anwendungssystem FlexNow[4] (Sinz und Wismans 1998) vorgestellt.

Aus dem Prüfungsprozess von Hochschulen heraus werden im Top-Down-Ansatz aus der Aufgabenebene die konkreten Rollen für ein Prüfungsverwaltungssystem modelliert und gezeigt, welche Konzepte aus eRBAC dabei Verwendung finden. Nach der Analyse der Aufgabenebene werden aus diesen die Zugriffsrechte und Rollen festgelegt.

1.3 Umfeld der Arbeit: FlexNow

An der Universität Bamberg wurde 1994 mit der Entwicklung von FlexNow begonnen, um den Prozess der Prüfungsverwaltung und die Serviceorientierung an Hochschulen zu unterstützen (Sinz 1995). Hintergrund bildete die zunehmende Einführung von studienbegleitenden Prüfungen.

Die einzelnen ausführbaren Programme von FlexNow, genannt Module, orientieren sich am Prüfungsprozess innerhalb einer Hochschule. Die Beschreibung der Module erfolgt, wie in Abb. 1-2 dargestellt, anhand eines Referenzprozesses für zentral organisierte Prüfungen. Die Prüfungsordnung bestimmt die Leistungselemente und deren Beziehungen bis zum Erreichen eines Abschlusses. Diese Typ-Informationen

[3] Zum Semantischen Objektmodell (SOM) siehe Ferstl und Sinz (2013, S. 194–236).

[4] FlexNow ist ein konfigurierbares Prüfungsverwaltungssystem mit webbasierter Selbstbedienungsfunktion für Studierende, das sich insbesondere für Bachelor- und Masterstudiengänge eignet.

stehen mit dem Erlass der Prüfungsordnung fest und werden im „Prüfungsordnungs“-Modul (PO-Modul) hinterlegt. Im Semesterablauf legt das Prüfungsamt im „Prüfungsdurchführungs“-Modul (PD-Modul), dem Leitstand von FlexNow, das Prüfungsangebot des jeweiligen Semesters fest. Nach der Veröffentlichung des Prüfungsangebotes erfolgt die webbasierte Anmeldung durch den Studierenden, wobei der individuelle Studienverlauf mit der Prüfungsordnung abgeglichen wird.

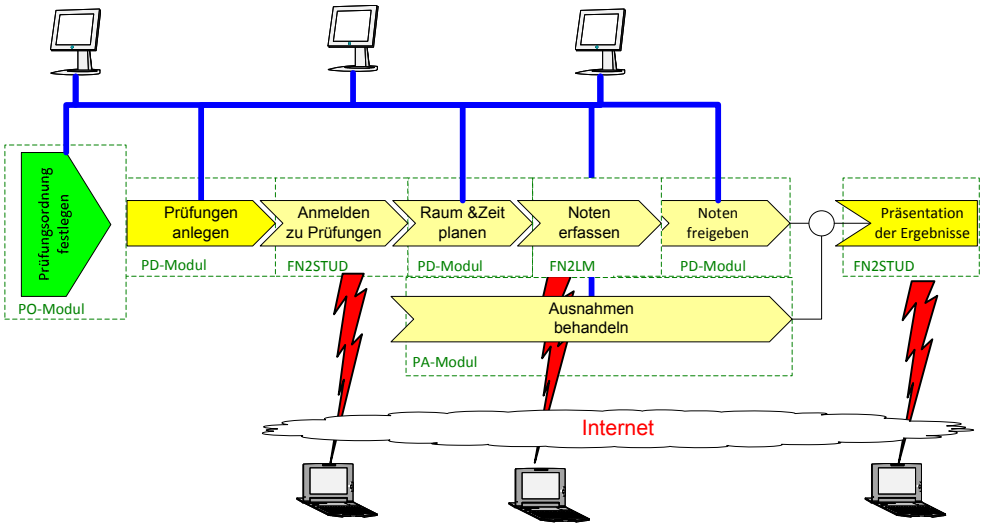

Abb. 1-2 Referenzprozess für zentral organisierte Prüfungen

Nach Ende des Anmelde- und Abmeldezeitraums wird mit dem PD-Modul die Raum- und Zeitplanung der schriftlichen und mündlichen Prüfungen vorgenommen. Nachdem die Prüfungen abgelegt sind, erfasst der Prüfer die Noten der Studierenden mit Hilfe des webbasierten „Lehrstuhl“-Moduls (FN2LM). Nach erfolgter Noteneingabe durch den Prüfer und Freigabe der Noten durch das Prüfungsamt können Studierende ihre Ergebnisse im Internet abrufen. Mit dem Prüfungsamts-Modul (PA-Modul) erhält außerdem ein Sachbearbeiter des Prüfungsamtes die Sicht auf einen einzelnen Studierenden, um gegebenenfalls Anerkennung, Studiengangwechsel und Ausnahmebehandlungen wie Krankheit hinterlegen zu können. Ergänzend erhält der Prüfungsausschussvorsitzende über das PA-Modul Einsicht in den Prüfungsverlauf eines Studierenden, um Informationen für seine Entscheidungen zu erhalten.

1.4 Aufbau der Arbeit

Die vorliegende Arbeit gliedert sich in neun Kapitel: In den Kapiteln 2 bis 4 werden neben der Basis der Informationssicherheit insbesondere Zugriffskontrollmodelle

und Konzepte für das rollenbasierte Zugriffskontrollmodell betrachtet. Kapitel 2 führt dazu in die Ziele, Grundlagen und Begriffe der Informationssicherheit und Zugriffskontrollsysteme ein. Zudem wird ein kurzer Überblick über die Authentifizierung einschließlich Kryptologie gegeben. Anschließend werden Zugriffskontrollstrategien vorgestellt und notwendige Eigenschaften von Zugriffskontrollmodellen und Konstruktionsprinzipen für sichere Zugriffskontrollsysteme festgelegt. In Kapitel 3 wird zunächst der Klassifikationsrahmen spezifiziert und anschließend Zugriffskontrollmodelle beschrieben. Mit der Einordnung im Klassifikationsrahmen wird die Eignung dieser für eine Zugriffskontrolle im IS untersucht, bewertet und geprüft inwieweit Konstruktionsprinzipien für sichere Zugriffskontrollsysteme erfüllt werden. Für das ausgewählte Referenzmodell des rollenbasierten Zugriffskontrollmodells (RBAC) werden in Kapitel 4 die Konzepte Gruppen- und Rollenkonzept, Aufgabentrennung, Rollenhierarchie, Administration, Delegation, negative Zugriffsrechte und Dynamisierung untersucht.

In den beiden folgenden Kapiteln werden die beiden Forschungsgebiete verbunden. In Kapitel 5 wird der Begriff Rolle durch die Einordnung in Rollenkonzepte strukturiert und zu einem ganzheitlichen Rollenkonzept entwickelt, das um AwS und Funktionen erweitert wird. Es wird anschließend untersucht, ob sich das Virtualisierungskonzept auf Rollen übertragen lässt und ob Rollen als virtuelle Aufgabenträger betrachtet werden können In Kapitel 6 fließen dann die Ergebnisse aus der Auswahl des Zugriffskontrollmodells, der Untersuchung der verschiedenen Konzepte und der Einordnung der Rolle ins betriebliche Informationssystem in die Erweiterung des rollenbasierten Zugriffskontrollmodells (eRBAC) ein.

Die beiden Kapitel 7 und 8 bilden den Proof of Concept. Kapitel 7 beschreibt die prototypische Realisierung von eRBAC: das Zugriffskontrollsystem FN2RBAC[5]. Dabei werden die Konzeption und Architektur beschrieben. Das Datenschema von FN2RBAC wird im Anhang beschrieben. In Kapitel 8 wird aus dem Geschäftsprozess „Universitätsprozess Prüfung“ die Zugriffskontrollstrategie ermittelt sowie eine konkrete Modellierung der Zugriffsrechte, Rollen und Rollenhierarchie als Fallstudie vorgestellt.

[5] FN2RBAC ist ein Akronym, das sich zusammensetzt aus FlexNow 2 (FN2) und der Abkürzung für role based access control (RBAC).

Das abschließende Kapitel 9 fasst die zentralen Ergebnisse dieser Arbeit zusammen und gibt einen Ausblick auf weiteren Forschungsbedarf im Bereich der Rollenmodellierung im Allgemeinen, im Umfeld der betrieblichen Informationssicherheit und der Zugriffskontrolle im Besonderen.

2 Informationssicherheit in Anwendungssystemen

Im vorliegenden Kapitel wird in die Grundlagen und Terminologie der Informationssicherheit mit dem Schwerpunkt Zugriffskontrolle einschließlich Konstruktionsprinzipien eingeführt. Daneben wird der Zusammenhang zwischen Authentifizierung, Zugriffskontrolle und Datenschutz analysiert. Es wird anschließend der Untersuchungsrahmen in Bezug auf Informationssicherheit sowie Zugriffskontrolle in Anwendungssystemen abgegrenzt. Die hier vorgestellten Grundlagen werden als Basis für die Beurteilung der Zugriffskontrollmodelle, für die Entwicklung des erweiterten rollenbasierten Zugriffskontrollmodells (eRBAC) und der Realisierung des Zugriffskontrollsystems FN2RBAC herangezogen.

Wichtige und schützenswerte Geschäftswerte sind Informationen (Eckert 2012, S. 4; Rockart et al. 1996, S. 53; Reichenbach 2004, S. 329f). Diese werden im Informationssystem eines Unternehmens oder einer Verwaltung erfasst, übertragen, transformiert, gespeichert und bereit gestellt (Ferstl und Sinz 2013, S. 3). Unternehmen und Verwaltungen sind von einem leistungsfähigen und funktionierenden Informationssystem abhängig (Junk und Mayer 2003, S. 5; Bundesamt für Sicherheit in der Informationstechnik (BSI) 2011, S. 1; International Organization for Standardization (ISO) 2005, S. viii). Informationssicherheit befasst sich mit der Sicherheit im Informationssystem und muss sicherstellen, dass es zu keiner unautorisierten Informationsveränderung oder –gewinnung kommt (Eckert 2012, S. 6; Reeg 2012, S. 20). Neben dem Schutz der Geschäftswerte müssen auch nicht kommerzielle Aspekte wie die Einhaltung gesetzlicher Vorschriften z. B. die Sicherstellung des Datenschutzes bei der Umsetzung der Informationssicherheit berücksichtigt werden (Linkies und Off 2006, S. 28).

Bei der Diskussion um Informationssicherheit darf nicht übersehen werden, dass es eine hundertprozentige Sicherheit nicht gibt (Landwehr et al. 1984, S. 198). Informationssicherheit „bezeichnet einen Zustand, in dem die Risiken, die beim Einsatz von Informationstechnik aufgrund von Bedrohungen und Schwachstellen vorhanden sind, durch angemessene Maßnahmen auf ein tragbares Maß reduziert sind“ (Bundesamt für Sicherheit in der Informationstechnik (BSI) 2011, S. 9). Durch die stetige Weiterentwicklung der Informationstechnologie können nicht alle in der Zukunft zu erwartenden Angriffe auf Informationssysteme erkannt und deren Auswirkungen

abgeschätzt werden. Informationssicherheit ist ein dynamisches Betrachtungsobjekt, die Entwicklung neuer Sicherheitsmechanismen folgt stets der technischen Entwicklung (Gasser 1988, S. 8; Sackmann 2012).

2.1 Grundlagen der Informationssicherheit

Die Grundlagen der Informationssicherheit umfassen Sachziele und Grundfunktionen sowie die Sicherheitsstrategie, ein Sicherheitskonzept auf strategischer Ebene, das die Sicherheitsziele eines Unternehmens festlegt (Sackmann 2012). Neben diesen Sachzielen gibt es zusätzlich externe und interne Vorgaben, die beim Erstellen der Sicherheitsstrategie berücksichtigt werden müssen. Nach einer Einordnung der Informationssicherheit in das IS wird der Untersuchungsrahmen abgegrenzt.

2.1.1 Ziele der Informationssicherheit

Im Bereich der Informationssicherheit werden Begriffe zum Teil uneinheitlich benutzt[6]. Als allgemeiner Konsens haben sich in der Literatur die nachstehenden vier Sachziele herauskristallisiert:

- Vertraulichkeit,
- Integrität,
- Verfügbarkeit,
- Verbindlichkeit (Pohl 2004, S. 679; Raepple 2001, S. 4).

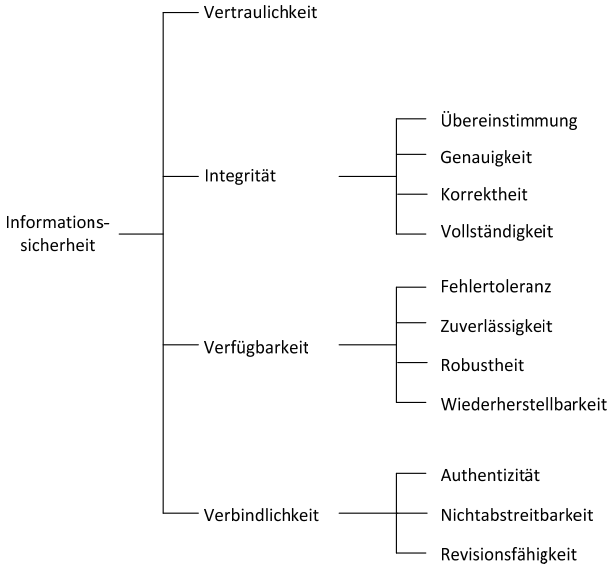

Abb. 2-1 Sachziele der Informationssicherheit mit ihren Komponenten nach (Pohl 2004, S. 680)

[6] Eine ausführliche Taxonomie der Begriffe findet sich in Pohl (2004); Schier (1999, S. 30–32); Reeg (2012, S. 32–36).

Vertraulichkeit ist der Schutz vor unberechtigtem Informationsgewinn, d. h. Informationen sind nur für Berechtigte zugreifbar. Dabei muss auch der Schutz während der Übertragung der Informationen über Netzwerke gewährleistet werden.

Die Sachziele Integrität, Verfügbarkeit und Verbindlichkeit unterteilen sich in weitere Komponenten, siehe **Abb. 2-1**.

Integrität stellt die Genauigkeit, Korrektheit und Vollständigkeit von Informationen und Verfahren sicher. Daten können nicht unberechtigt verändert, gelöscht oder zerstört werden. Es muss eine Übereinstimmung zwischen tatsächlichem Wert eines Objektes und dem verarbeitenden bzw. gespeicherten Wert bestehen. Integrität setzt sich zusammen aus:

- **Systemintegrität**. Sie bezieht sich auf eine korrekte Funktionsweise eines AwS, d. h. Ist- und Soll-Funktionalität stimmen überein. Diese Funktionssicherheit ist die Voraussetzung für Datenintegrität.
- **Datenintegrität**. Sie umfasst die Sicherstellung der Korrektheit, also Unversehrtheit von Daten. Die Daten müssen vor der unberechtigten Modifikation geschützt werden.

Verfügbarkeit stellt sicher, dass alle benötigten Daten sowie die zur Verarbeitung notwendigen Anwendungssysteme und Betriebsmittel jederzeit verfügbar und funktionsbereit sind, wenn ein autorisierter Nutzer zugreifen will. Die Komponenten der Verfügbarkeit sind: Zuverlässigkeit, Fehlertoleranz, Robustheit und Wiederherstellbarkeit. Diese müssen durch die Funktionssicherheit des AwS gewährleistet werden.

Verbindlichkeit ist die Eigenschaft eines Systems, zurechenbare und rechtsverbindliche Kommunikation zu unterstützen. Sie schützt den Sender gegen Täuschung und damit gibt es keine Möglichkeit des Abstreitens durch Sender oder Empfänger. Durch Verbindlichkeit werden in einem Informationssystem alle versuchten und erfolgten Zugriffe von Subjekten[7] auf Objekte nachvollziehbar beschrieben. Dabei werden Zugriffe Subjekten zugeordnet und es wird die Erkennung und Untersuchung von Angriffen ermöglicht. Grundlage für die Verbindlichkeit sind Authentizität des Subjekts sowie Revisionsfähigkeit der Protokollierung der Zugriffe (Eckert 2012, S. 12–13; Pohl 2004, S. 679–680).

[7] Die Begriffe Nutzer, Subjekt, Objekt werden neben weiteren Begriffen in Abschnitt 2.4.2.2 ausführlich beschrieben.

Im Kontext der Informationssicherheit von Anwendungssystemen ist die hinreichende Erfüllung der Sachziele Vertraulichkeit, Integrität und Verbindlichkeit sicherzustellen und setzt voraus, dass die Funktionssicherheit der Anwendungssoftware gewährleistet ist (Eckert 2012, S. 6). Zusätzlich muss die Verfügbarkeit durch die Installation, die Einbettung in die Rechnerarchitektur und geeignete Notfallkonzepte sichergestellt werden.

2.1.2 Grundfunktionen der Informationssicherheit

Die Erfüllung der Sachziele der Informationssicherheit wird durch folgende Grundfunktionen erreicht:

- Authentifizierung,
- Rechteverwaltung,
- Rechteprüfung,
- Protokollierung (Audit).

Um unerlaubte Zugriffe auf ein System zu verhindern, müssen Subjekte identifizierbar sein. Die Authentifizierung übernimmt die Identitätsprüfung eines Subjektes und stellt seine vorgegebene Identität sicher (Eckert 2012, S. 8–9; Seufert 2001, S. 11). Sie überprüft die Echtheit einer Person, einer Organisation oder eines Programms. Die Authentifizierung legt jedoch nicht fest, auf welche Daten und Funktionen ein Subjekt zugreifen darf. Das Verfahren der Authentifizierung[8] muss vom Zugriffskontrollsystem vollständig getrennt sein und ist diesem zwingend vorgelagert. Sie ist damit eine notwendige Voraussetzung für eine funktionierende Zugriffskontrolle (Essmayr et al. 2004, S. 132).

Die Rechteverwaltung bildet die Datenbasis für die Rechteprüfung zur Abwehr von unautorisierten Zugriffen auf Objekte. Sie speichert Informationen über die Subjekte und zu schützenden Objekte mit ihren Operatoren. Die Rechteverwaltung hat sicherzustellen, dass alle Subjekte und Objekte eindeutig und fälschungssicher identifiziert werden und dass jedes Objekt, für das Zugriffsbeschränkungen festgelegt sind, auch von der Rechteverwaltung erfasst wird (Eckert 2012, S. 219, 627-628).

Die Rechteprüfung kontrolliert bei jedem Zugriff, ob ein Subjekt **s** das Objekt **o** mit dem Operator **op** aufrufen darf und stützt sich auf die gespeicherten Daten der

[8] Eine Einführung in die Authentifizierung und einen Überblick möglicher Verfahren wird in Kapitel 2.3 gegeben.

Rechteverwaltung. Sie prüft dabei ob Funktionen von einem Subjekt aufgerufen und ausgeführt werden dürfen. „Zwischen der Rechteprüfung und der Ausübung des Rechts sollte keine Aktion möglich sein, die den Rechteentzug zur Folge hat“ (Eckert 2012, S. 628). Die Rechteprüfung ist der Authentifizierung nachgeschaltet. Ist die Authentifizierung nur unzureichend gewährleistet, wird auch die Rechteprüfung als gescheitert angesehen.

Alle Zugriffe und ausgeführten Funktionen sollten protokolliert werden, um eine nachträgliche Analyse der Zugriffe zu ermöglichen. Nach der Rechteprüfung wird sofort eine Protokollierung vorgenommen, so dass die Verbindlichkeit hergestellt werden kann. Für die Beweissicherung legen die Sicherheitsanforderungen fest, welche Ereignisse und Informationen zu protokollieren sind. Die Identität eines Subjektes, die aufgerufenen Objekte mit den dazugehörigen Operatoren und der dazugehörige Zeitpunkt einer Aufrufanforderung sind mindestens zu protokollieren (Eckert 2012, S. 220–221).

2.1.3 Einflussfaktoren auf die Sicherheitsstrategie

Basis für das Ergreifen von Maßnahmen zur Erhaltung der Informationssicherheit ist die Sicherheitsstrategie. Unter Berücksichtigung der strategischen Sicherheitsziele bietet eine Risikoanalyse die Grundlage für die praktische Umsetzung. Diese untersucht, wie wahrscheinlich das Eintreten eines schädigenden Ereignisses ist und welche negativen Folgen sich daraus für Verfügbarkeit, Integrität, Vertraulichkeit und Verbindlichkeit ergeben. Der IT-Grundschutzkatalog liefert das für Unternehmen und Verwaltungen notwendige praktische Wissen und die Formblätter für die Umsetzung der Grundfunktionen (Bundesamt für Sicherheit in der Informationstechnik (BSI) 2011, S. 7).

Den Weg von der Risikoanalyse und externen Vorgaben bis zur Sicherheitsstrategie stellt **Abb. 2-2** dar. Die Ergebnisse einer Bedrohungs- und Risikoanalyse bestimmen die Sicherheitsanforderungen innerhalb von Unternehmen und Verwaltungen. Diese Sicherheitsanforderungen legen Forderungen und Regeln fest, wie sicherheitskritische Informationen zu behandeln sind und definieren die Schwerpunkte bei der Erfüllung der Sachziele. Daraus resultiert die Sicherheitsstrategie, die informell oder präzise formalisiert beschrieben werden kann (Eckert 2012, S. 216–217; Kersten 1993, S. 615).

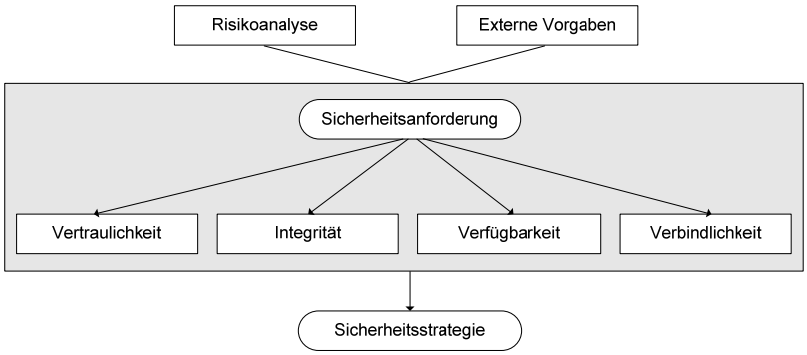

Abb. 2-2 Von der Sicherheitsanforderung zur Sicherheitsstrategie nach (Seufert 2001, S. 30)

Neben den Ergebnissen der Risikoanalyse beeinflussen externe Vorgaben die Sicherheitsanforderungen. Externe Vorgaben sind gesetzliche Vorgaben, aufsichtsrechtliche und regulatorische Anforderungen, nationale und internationale Standards sowie kunden- und unternehmensseitige Anforderungen (Reichenbach 2004, S. 331). **Tab. 2-1** zeigt ausgewählte Anforderungen ohne Anspruch auf Vollständigkeit.

Tab. 2-1 Auswahl externer Vorgaben für Sicherheitsanforderungen (Reichenbach 2004, S. 336–338)

Externe Vorgaben	Beispiele für externe Vorgaben
Gesetzliche Vorgaben	• Handelsgesetzbuch • Bundesdatenschutzgesetz • Informations- und Kommunikationsdienste-Gesetz • Gesetz zur Kontrolle und Transparenz im Unternehmensbereich • Gesetzliche Kryptoregulierungen
Aufsichtsrechtliche und regulatorische Anforderungen	• Grundsätze ordnungsmäßiger DV-gestützter Buchführungssysteme • Grundsätze ordnungsmäßiger Buchführung bei Einsatz von Informationstechnologie – Stellungnahme FAIT1. • Grundsätze ordnungsmäßiger Buchführung bei Einsatz von Electronic Commerce FAIT2 • Verlautbarungen der Bundesanstalt für Finanzdienstleistungsaufsicht
Nationale Standards	• IT-Grundschutzhandbuch • IT-Sicherheitshandbuch des Bundesamtes für Sicherheit in der Informationstechnik
Internationale Standards	• ISO 17799: Information Security Management 2004 • International Standards Organization / International Electrotechnical Commision Joint Technical Committee 1 Subcommittee 27 – ISO/IEC JTC 1 CS 27 – Standards für IT-Sicherheit • National Institute of Standards and Technology: An Introduction to Computer Security: The NIST Handbook, Special Publication 800-12, 1995
Unternehmensseitige Anforderungen	• Sicherung von Betriebsvermögen und interner Informationen • Vermeidung von Vertrauensschäden
Kundenseitige Anforderungen	• Vertraulichkeit von Vermögensverhältnissen (Bankgeheimnis) • Schutz von Vermögenswerten

In einem Prüfungsverwaltungssystem sind beispielsweise von den in ***Tab. 2-1*** *aufgeführten externen Vorgaben besonders die verschiedenen Datenschutzgesetze zu berücksichtigen, da persönliche Daten und der Studienverlauf eines Studierenden gespeichert werden.*

Im Abschnitt 2.2 wird deshalb auf den Datenschutz eingegangen und sein Einfluss auf die Anforderungen der Informationssicherheit für Anwendungssysteme herausgearbeitet.

2.1.4 Einordnen der Informationssicherheit ins betriebliche Informationssystem

Ein IS in Wirtschaft und Verwaltung ist das gesamte informationsverarbeitende Teilsystem (Ferstl und Sinz 2013, S. 4). Es beinhaltet eine Menge von Informationsverarbeitungsaufgaben und Aufgabenträger. Das IS kann unterteilt werden in Aufgabenebene und Aufgabenträgerebene. Die Menge aller Aufgaben zusammen mit den Informationsbeziehungen bilden die Aufgabenebene eines IS. Die Aufgabenträgerebene wird durch die Menge aller maschinellen und personellen Aufgabenträger gebildet, die durch Kommunikationskanäle verbunden sind (Ferstl und Sinz 2013, S. 4f).

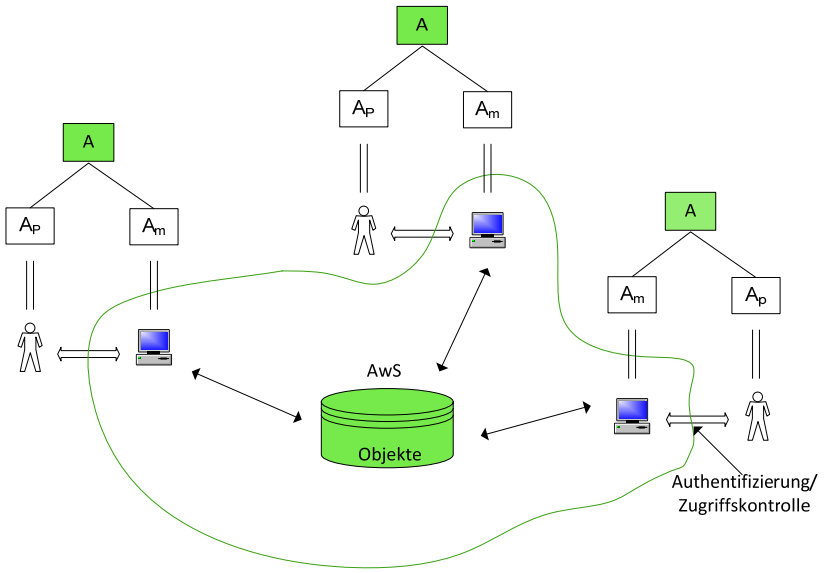

Abb. 2-3 Teilautomatisierte Aufgaben im IS

Teilautomatisierte Aufgaben (A) eines betrieblichen Informationssystems bestehen aus einem automatisierbaren (A_m) und einem nicht automatisierbaren (A_p) Aufgabenanteil. Die Durchführung einer teilautomatisierten Aufgabe erfordert auf der

Aufgabenträgerebene eine Kooperation von personellen und maschinellen Aufgabenträgern (Ferstl und Sinz 2013, S. 58). Der automatisierte Teil der Aufgaben wird von maschinellen Aufgabenträgern durchgeführt, die zu einem integrierten Anwendungssystem (AwS) zusammengefasst werden. Die Objekte des AwS sind vor unerlaubten Zugriffen durch ein Zugriffskontrollsystem zu schützen.

Zur Verrichtung teilautomatisierter Aufgaben ist eine Mensch-Computer-Kommunikation erforderlich, da ein personeller Aufgabenträger mit einem maschinellen Aufgabenträger „kommuniziert". **Abb. 2-3** zeigt, dass an dieser Schnittstelle zur Gewährleistung der Informationssicherheit eine Authentifizierung und Zugriffskontrolle durchgeführt werden muss. Bei einer Mensch-Computer-Interaktion muss sichergestellt werden, dass nur Objekte verfügbar sind, die ein personeller Aufgabenträger zur Erledigung seiner Aufgaben benötigt und bearbeiten darf.

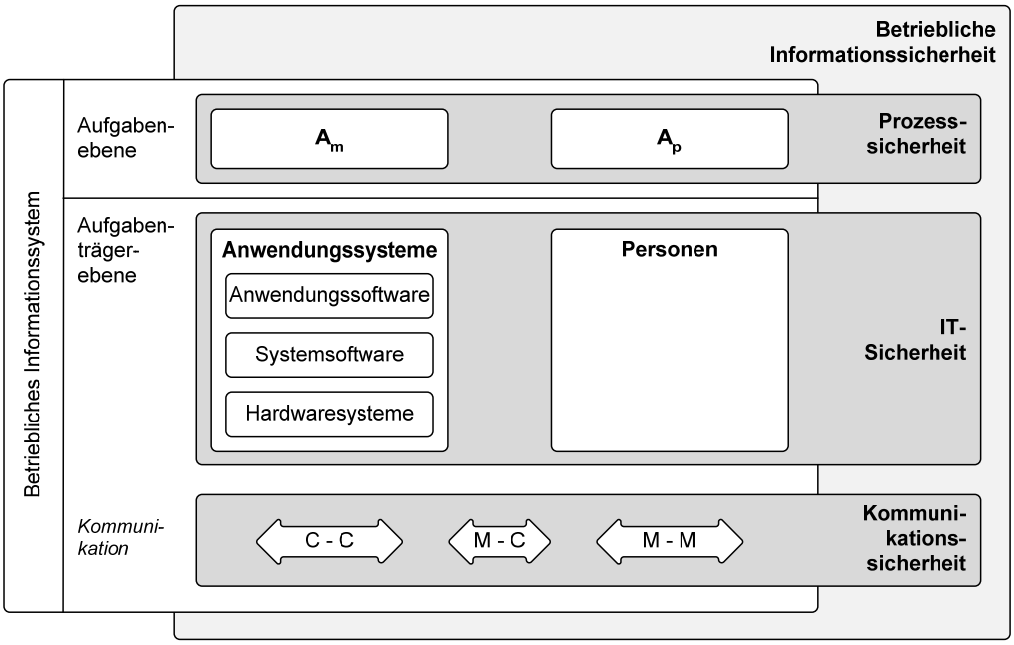

Abb. 2-4 Teilbereiche und Einordnung betrieblicher Informationssicherheit (Reeg 2012, S. 27)

Die Informationssicherheit im betrieblichen Informationssystem wird unter dem Begriff betriebliche Informationssicherheit beschrieben (Reeg 2012, S. 25). **Abb. 2-4** zeigt eine Einordnung der betrieblichen Informationssicherheit in das IS. Sie kann in die Teilbereiche: Prozesssicherheit, IT-Sicherheit und Kommunikationssicherheit unterteilt werden. Prozesssicherheit bezieht sich auf die Aufgabenebene, die Kommunikationssicherheit auf die Kommunikationssysteme der Aufgabenträger. Die IT-Sicherheit beschäftigt sich mit der Aufgabenträgerebene an der Schnittstelle zwischen maschinellen und personellen Aufgabenträgern (Reeg 2012, S. 26–28). Die

vorliegende Arbeit betrachtet den Bereich der IT-Sicherheit und hier insbesondere die Zugriffskontrolle für Anwendungssoftware an der Schnittstelle zwischen personellen und maschinellen Aufgabenträgern. Eine Herausforderung bei der Gewährleistung der IT-Sicherheit in der Anwendungssoftware ist die höhere Komplexität der Zugriffskontrolle im Gegensatz zur Zugriffskontrolle in Betriebssystemen (Kern et al. 2004, S. 88f).

2.2 Datenschutz

Dieses Kapitel beschäftigt sich mit der übergeordneten Anforderung des Rechts auf informationelle Selbstbestimmung, dem Datenschutz und insbesondere dem Bundesdatenschutzgesetz (BDSG) und seine Auswirkungen auf die Informationssicherheit (Schier 1999, S. 32–33). Datenschutz ist „ein schlagwortartiger Begriff mit dem die Gesamtheit der Rechtsregeln bezeichnet wird, die dem Schutz des Persönlichkeitsrechts dienen“ (Ehmann 1993, S. 74). Das BDSG beinhaltet nicht alle Rechtsregeln im Bereich des Datenschutzes (Ehmann 1993, S. 74), sondern die darin aufgeführten Regelungen gelten allgemein und sind im Verhältnis zu bereichsspezifischen Vorschriften wie dem Gesetz für Telekommunikation des Bundes (TKG), dem Teledienstedatenschutzgesetz (TDDSG) und dem Mediendienste-Staatsvertrag (MDStV) subsidiär (Tinnefeld et al. 2005, S. 641–643).

Bezogen auf die Sachziele der Informationssicherheit zielt das BDSG auf das Sachziel Vertraulichkeit ab. Das BDSG wird immer wieder, auch an europäische Richtlinien, angepasst. Die letzte Änderung als Gesetz wurde am 14.08.2009 verabschiedet, danach wurde es 2010 neu gefasst und aktualisiert (bfdi 2010). Das BDSG gilt für öffentliche Stellen des Bundes und der Länder sowie nicht öffentliche Stellen soweit diese Daten unter Einsatz von automatisierter Datenverwaltung oder durch Ablage in strukturierten Dateien gespeichert werden. Für nicht öffentliche Stellen gilt das BDSG nur, wenn die Daten nicht für private Zwecke verwendet werden.

§1 Abs. 1 legt den Zweck des BDSG fest: „Zweck dieses Gesetzes ist es, den einzelnen davor zu schützen, dass er durch den Umgang mit seinen personenbezogenen Daten in seinem Persönlichkeitsrecht beeinträchtigt wird“ (bfdi 2010).

Der Datenschutz[9] kann nicht isoliert von der Informationssicherheit betrachtet werden. **Abb. 2-5** zeigt, wie beide ineinander greifen. Datenschutz schützt natürliche Personen und soll die Verletzung von Persönlichkeitsrechten durch die Speicherung und Auswertung von persönlichen Daten verhindern. Informationssicherheit hat die betrieblichen Informationen im Fokus. Damit soll der Verlust bzw. die Zerstörung von gespeicherten Informationen sowie der Missbrauch durch Unbefugte verhindert werden. Anlage 1 des § 9 des BDSG schreibt acht Kontrollen und Maßnahmen vor, um vor einer Verletzung des Persönlichkeitsrechts zu schützen.

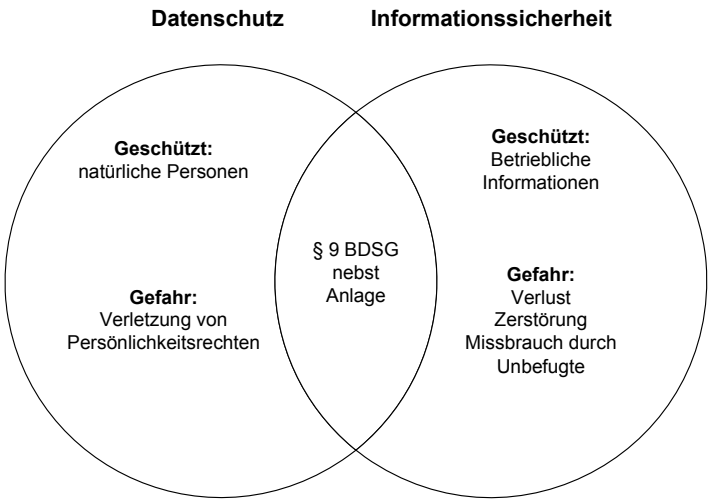

Abb. 2-5 Abgrenzung von Datenschutz und Informationssicherheit nach (Gola und Jaspers 2002, S. 14)

Die Gegenüberstellung der Maßnahmen und der zur Implementierung notwendigen Grundfunktionen der Informationssicherheit in **Tab. 2-2** zeigt, dass die vorgeschriebenen Kontrollen in Anlage 1 zu § 9 des Datenschutzgesetzes die Informationssicherheit im Bereich der Speicherung persönlicher Daten ergänzen. Obwohl die allgemeinen Ziele der Informationssicherheit Integrität, Verfügbarkeit und Verbindlichkeit sowie deren Grundfunktionen im BDSG nicht explizit genannt werden, wirkt die technische Umsetzung des Datenschutzes indirekt über die acht Kontrollbereiche auf die Grundfunktionen zur Gewährleistung der Informationssicherheit.

Das Datenschutzgesetz muss bei der Modellierung und Realisierung der Rechteverwaltung, der Rechteprüfung und der Protokollierung berücksichtigt werden. Der Datenschutz verstärkt das Sachziel Vertraulichkeit und ergänzt die Sachziele Integrität und Verbindlichkeit. Er hat damit Einfluss auf die Implementierung der

[9] Weiterführende Beschreibungen und Diskussionen zum Datenschutz finden sich in Gola und Jaspers (2002); bfdi (2010); Tinnefeld et al. (2005)

Grundfunktionen der Informationssicherheit. Auch im Bereich des Prüfungswesens müssen persönliche Daten, wie Adressen oder Ergebnisse von Prüfungen des Studierenden geschützt werden. Beispielhaft wird das Trennungsgebot herausgegriffen, um die Auswirkung auf den Prüfungsprozess zu verdeutlichen.

Ein Prüfer darf nur Leistungsnachweise von Studierenden einsehen, für deren Prüfungen er verantwortlich ist. Alle Noten können dennoch physikalisch in derselben Tabelle abgespeichert werden. Das Trennungsgebot kann über eine Anzeige auf fachlicher Ebene durch das Zugriffskontrollsystem und/oder AwS überprüft und gewährleistet werden.

Tab. 2-2 § 9 Anlage 1, die Kontrollbereiche mit ihren Maßnahmen und Gegenüberstellung zu Grundfunktionen der Informationssicherheit (Gola und Jaspers 2002, S. 41; Tinnefeld et al. 2005, S. 661–665; Voßbein 2005, S. 13)

Kontrollen lt. Anlage § 9	Maßnahmen, die dies realisieren	Grundfunktionen der Informationssicherheit
Zutrittskontrolle	Physikalischen Zugang zum Gelände sichern	Authentifizierung
Zugangskontrolle	Passwortschutz und Verschlüsselung	Authentifizierung
Zugriffskontrolle	Firewall-Systeme, Passwortschutz, Zugriffskontrolle, Berechtigungskonzepte	Authentifizierung, Rechteverwaltung, Rechteprüfung und Protokollierung
Weitergabekontrolle	Verschlüsselung, Protokollierung, Regelung des Kommunikationsverkehrs	Authentifizierung, Rechteverwaltung, Rechteprüfung und Protokollierung
Eingabekontrolle	Sicherungssoftware, Berechtigungskonzept	Protokollierung
Auftragskontrolle	Weisungen des Auftraggebers, Protokollierung	Rechteverwaltung, Rechteprüfung und Protokollierung
Verfügbarkeitskontrolle	Sicherheitskopien an anderen Orten, Maßnahmen zum Katastrophenschutz	./.
Trennungsgebot	logische, keine physikalische Trennung, Benutzerprofile, Berechtigungen	Rechteverwaltung, Rechteprüfung und Protokollierung

Nach der Einführung in die Informationssicherheit und einem kurzen Überblick über den Datenschutz wird im Weiteren auf die Grundfunktion der Authentifizierung eingegangen. Sie ist der Zugriffskontrolle vorgeschaltet und muss durch die Systemarchitektur sichergestellt werden (Essmayr et al. 2004, S. 129).

2.3 Authentifizierung

Unter Authentizität eines Subjekts wird die Echtheit und Glaubwürdigkeit verstanden, die anhand seiner Identität und seiner charakterisierenden Eigenschaften überprüfbar ist. Mit Authentifizierung wird ein Subjekt mit einer eindeutigen Kennung

verbunden. Die Authentifizierungsinformation, z. B. Passwörter, muss von den Identifikationsinformationen getrennt aufbewahrt werden, da Passwörter geheim, Kennungen aber öffentlich sind (Gasser 1988, S. 23). Authentifizierung überprüft mit geeigneten Methoden die Korrektheit der behaupteten Identität eines Gegenübers und ermittelt damit die Quelle einer Anfrage.

Die Authentifizierung kann mit verschiedenen Merkmalen durchgeführt werden. In der Praxis werden zur Authentifizierung folgende drei Merkmale verwendet:

- Kenntnisse eines spezifischen Wissens (Passwort),
- persönlicher Besitz (Chipkarte),
- Biometrie: Überprüfung eines bestimmten Merkmals z. B. körperliches Merkmal (Fingerabdruck) oder Verhaltens z. B. typische Bewegungsmuster (Tastenanschlag).

Darüber hinaus existieren Kombinationen der oben genannten Merkmale sog. Mehr-Faktor-Authentifizierung, beispielsweise eine Chipkarte (Besitz) zusammen mit einem PIN (Passwort) oder ein Ausweis zusammen mit einem biometrischen Merkmal. Damit werden die Vorteile unterschiedlicher Techniken kombiniert (Eckert 2012, S. 461; Müller 2005, S. 173–176).

Es gibt verschiedene Verfahren, die eine Authentifizierung mit Hilfe von Wissen realisieren. Diese sind unter anderem Passwort, PIN, Einmalpasswort, Challenge-Response und One Time PIN Token (Eckert 2012, S. 462–485). Bekannteste Vertreter der Verfahren, die einen Besitz überprüfen, sind Chipkarten, daneben existieren noch Magnetstreifenkarte, Krypto-Token, Schlüssel, RFID-Karte und Zertifikate[10] (Müller 2005, S. 173). Biometrie bezeichnet zusammenfassend die Überprüfung persönlicher Merkmale oder persönlichen Verhaltens. Beispiele sind Fingerabdruck, Geschichtserkennung, Tastaturanschlag, DNA, Stimme, Handlinienstruktur und Augen-Netzhaut.

Im folgenden Abschnitt werden ausgewählte Authentifizierungsverfahren vorgestellt: Kryptografie wird als grundlegende Technik für eine sichere Authentifizierung benötigt[11]. Daran schließt sich die Beschreibung des Passwort- und Challenge-

[10] Eine verteilte Authentifizierung über Zertifikate wird in dieser Arbeit nicht behandelt, dazu wird auf die entsprechende Literatur verwiesen Rieger (2007).

[11] Für ein vertieftes Studium der benötigten Techniken, wie Hashfunktionen, Kryptoalgorithmen, digitale Signatur, Zertifikate wird auf Eckert (2012, S. 299–459); Rankl und Effing (2002, S. 153–232); Schwenk (2005); Hühnlein und Korte (2006, S. 21–88) verwiesen.

Response-Verfahrens zur Authentifizierung an. Danach wird ein kurzer Überblick über die Technik der Chipkarten und eine Erläuterung der Verfahren, die für eine Authentifizierung in diesem Bereich notwendig sind, gegeben. Abschließend werden Verfahren der Authentifizierung durch Biometrie beschrieben. Eine kurze Beschreibung des Single-Sign-On-Verfahrens schließt dieses Kapitel ab.

2.3.1 Kryptografie

Damit das Beschaffen von Passwörtern erschwert wird, müssen diese verschlüsselt gespeichert und übertragen werden. Das Forschungsgebiet, dass sich mit der Verschlüsselung beschäftigt ist die Kryptografie. Ein kryptografisches System legt fest, wie Klartexte mit einem Kryptoalgorithmus in Kryptotexte transformiert werden und wie Kryptotexte entschlüsselt werden, um wieder zu den Ausgangstexten zu gelangen.

Die Kryptoalgorithmen lassen sich unterteilen in:

- symmetrische und
- asymmetrische Verfahren.

Bei symmetrischen Verfahren wird zur Ver- und Entschlüsselung der gleiche Schlüssel verwendet. Da zwischen Partnern die Kommunikation verschlüsselt mit demselben gemeinsamen und geheimen Schlüssel stattfindet, wird von symmetrischen Schlüsseln gesprochen. Dafür muss ein Austausch des gemeinsamen Schlüssels vor der Kommunikation über einen sicheren Kanal zwischen den Partnern erfolgen. Die Sicherheit eines Authentifizierungsverfahrens, das die Verschlüsselung mit einem symmetrischen Schlüssel vornimmt, hängt neben einer sicheren Aufbewahrung vom sicheren Austausch des Schlüssels ab. Ein weiterer Nachteil dieses Verfahrens ist, dass für jeden Kommunikationspartner ein eigener Schlüssel vorhanden sein muss, dadurch ergibt sich eine große Anzahl von Schlüsseln (Eckert 2012, S. 308; Eren und Detken 2006, S. 160–161).

Exkurs Hashfunktion und Einwegfunktion: Eine Funktion $y = f(x)$ *heißt Einwegfunktion, wenn sich für alle Funktionswerte* $x \in X$ *der Funktionswert* $f(x)$ *effizient berechnen lässt, aber für die Berechnung des Ausgangswertes* $x = f^{-1}(y)$ *kein effizientes Verfahren existiert und dafür ein sehr großer Rechenaufwand benötigt*

wird (Eckert 2012, S. 349; Hühnlein und Korte 2006, S. 22)[12]. Hashfunktionen sind eine spezielle Klasse von Einwegfunktionen und bilden eine Zeichenfolge beliebiger Länge auf eine Zeichenfolge fester Länge, dem Hashwert, ab (Bless et al. 2005; Buchmann 2004, S. 191; Reeg 2012, S. 129).

Das asymmetrische Verfahren[13] verwendet für die Ver- bzw. Entschlüsselung unterschiedliche Schlüssel. Jeder Kommunikationspartner besitzt ein Schlüsselpaar, einen privaten, geheimen und einen öffentlichen Schlüssel. Es gilt, was mit dem öffentlichen Schlüssel verschlüsselt wird, kann ausschließlich mit dem dazugehörigen privaten Schlüssel entschlüsselt werden. Den privaten geheimen Schlüssel kennt nur der Besitzer und er wird nicht weitergegeben, während der öffentliche Schlüssel jedermann zugänglich ist und auch über unsichere Kanäle verteilt werden kann. Eine Bedingung dabei ist, dass sich aus dem öffentlichen Schlüssel nicht der private Schlüssel errechnen lässt. Für die Berechnung des öffentlichen Schlüssels aus dem privaten Schlüssel wird eine Einwegfunktion verwendet. Ein asymmetrisches Verfahren bietet zwei Vorteile: Es verringert sich die Anzahl der Schlüssel für den Austausch zwischen beliebigen Partnern und es wird die Implementierung einer elektronischen Unterschrift ermöglicht. Nachteil dieses Verfahrens ist, dass der Aufwand für Ver- bzw. Entschlüsselung höher ist als bei symmetrischen Verfahren und daher für große Datenmengen und Echtzeitverfahren nur bedingt geeignet ist (Schäfer 2003, S. 60; Eren und Detken 2006, S. 160–161; Eckert 2012, S. 309). Das symmetrische und asymmetrische Verfahren können auch gemeinsam angewandt werden (Rankl und Effing 2002, S. 180).

Folgende Anforderungen sind an sicherere Kryptoalgorithmen zu stellen:

- Kryptoalgorithmen sollten auf dem Kerckhoffs'-Prinzip basieren. Das Prinzip besagt, dass die gesamte Sicherheit eines Algorithmus auf der Geheimhaltung der Schlüssel beruhen soll und nicht auf der Geheimhaltung des kryptografischen Algorithmus (Kerckhoffs 1883).
- Die Schlüssellänge muss so groß gewählt werden, dass der Schlüssel nicht mit vertretbarem Aufwand ermittelt werden kann. Da die Rechenkapazität stetig steigt, werden immer größere Schlüssel erforderlich. Die empfohlenen

[12] Dass Einwegfunktionen nicht nur notwendig, sondern auch hinreichend sichere Signaturverfahren sind, zeigte Rompel (1990).

[13] Das asymmetrische Verfahren wurde unabhängig voneinander von Diffie und Hellman (1976) und Merkle (1978) entwickelt.

Schlüssellängen werden von der Bundesnetzagentur jährlich neu veröffentlicht[14] (Eckert 2012, S. 312).

- Eine seit den 90er Jahren hinzu genommene Anforderung ist die Rauschfreiheit, die besagt, dass die Ausführungszeit unabhängig von Schlüssel, Klartext und Schlüsseltext sein muss (Rankl 2008, S. 145).

Wie beim Thema Informationssicherheit existiert auch bei der Kryptografie keine absolute Sicherheit. Es wird hierbei zwischen theoretischer bzw. absoluter und praktischer Sicherheit unterschieden. Ein System wird als theoretisch sicher bezeichnet, wenn ein Angreifer unbegrenzt Zeit und Mittel zur Verfügung hat und die Verschlüsselung dennoch nicht brechen kann. Ein System wird als praktisch sicher bezeichnet, wenn ein Angreifer nur begrenzte Zeit und Hilfsmittel hat und damit die Verschlüsselung nicht brechen kann (Eckert 2012, S. 312: 316-319; Rankl und Effing 2002, S. 181).

2.3.2 Authentifizierung mittels Wissen

Methoden zur Authentifizierung auf Grundlage der Kenntnis eines spezifischen Wissens z. B. eines Passwortes sind in der Praxis am häufigsten anzutreffen und werden im Folgenden vorgestellt. Ausgewählte Implementierungsmöglichkeiten werden im Anhang beschrieben.

2.3.2.1 Passwortverfahren

Ein Subjekt authentifiziert sich mittels eines Passwortes, indem ein Geheimnis ausgetauscht wird. Das Passwortverfahren findet Verwendung an einem Arbeitsplatzrechner, zwischen Arbeitsplatzrechner und Server oder in einem Netzwerk. Das System, das eine Authentifizierung vornimmt, hat die Passwörter sicher zu verwalten. Es werden dafür kryptografische Verfahren eingesetzt.

Eine sichere Speicherung der Passwörter reicht für die Sicherheit des Passwortverfahrens nicht aus. Die Sicherheit hängt neben der sicheren Speicherung von der Zusammensetzung des gewählten Passwortes[15] ab. Beide Maßnahmen zur Sicherung der Authentifizierungsinformationen sind nutzlos, wenn ein Angreifer auf einfache

[14] Ein Schlüssel für symmetrische Verfahren sollte eine Länge von mindestens 128 Bit besitzen; dieser gilt bis zum Jahr 2036 als sicher. Für asymmetrische Verfahren werden 2048 Bit empfohlen, um eine langfristige Sicherheit sicherzustellen (Eckert 2012, S. 312)

[15] Die Anforderungen, um einen Mindeststandard zu gewährleisten finden sich u.a. in (Eckert 2012, S. 465–466).

Weise in den Besitz des Passwortes kommen kann. Um vom Nutzer festgelegte Passwörter zu umgehen, können vom System generierte Passwörter verwendet werden, welche jedoch den Nachteil besitzen, dass sie für den Nutzer schwer zu merken sind (Eckert 2012, S. 464–466).

Nach der Authentifizierung am Arbeitsplatzrechner kann bei der Weitergabe der Passwörter in Netzwerken das erweiterte Konzept der Einmal-Passwörter (siehe Kapitel 2.3.2.2) oder bei der Authentifizierung von Chipkarten das Challenge-Response-Verfahren (siehe Kapitel 2.3.2.3) verwendet werden.

2.3.2.2 Einmal-Passwörter

Bei der Anmeldung in Netzwerken existiert gegenüber der klassischen Vorgehensweise der Weiterreichung des vom Nutzer eingegebenen Passwortes das Verfahren des einmal nutzbaren Passwortes (engl. One-time Password). (Haller 1994) stellte 1994 die Funktionsweise seines entwickelten S/Key™[16] Verfahrens vor, dessen Algorithmus frei verfügbar ist. Das Verfahren soll Passwort-Sniffer-Angriffe in Netzwerken wirkungslos machen[17].

Ausgangspunkt des Verfahrens ist ein geheimes Passwort **s**, das zwischen dem Subjekt und seinem Arbeitsplatzrechner (Client) vereinbart wurde. Ein Server, auf den zugegriffen werden soll, benötigt dieses vom Nutzer eingegebene Passwort nicht. Aus diesem geheimen Passwort wird durch mehrmaliges anwenden einer kryptografisch sicheren Einwegfunktion **f** eine Liste von Einmalpasswörtern generiert. Die Sicherheit des Verfahrens beruht auf der Sicherheit der Einwegfunktion.

Das erste Passwort p_0 wird auf dem Arbeitsplatzrechner durch **N**-maliges Anwenden der Einwegfunktion **f** auf dem Geheimnis **s** erzeugt.

$$p_0 = f_N(s)$$

Das nächste Passwort wird generiert, in dem die Einwegfunktion **N-1**-mal ausgeführt wird.

$p_1 = f_{N-1}(s)$ => allgemein $p_i = f_{N-i}(s)$.

[16] Der Name S/Key ist ein eingetragener Handelsname der Firma Bellcore. Die Norm RFC 2289 spricht vom One Time Password (OTP).

[17] Für ein ausführliches Studium der technischen Umsetzung wird auf die Literatur verwiesen Haller (1994); Haller et al. (1998); Eckert (2012, S. 467–469).

Der Initialwert der Einmalpasswörter p_0 wird zusammen mit **i = 0** an den Server übermittelt. Dieser Initialwert wird auf dem Server gespeichert. Will sich ein Client authentifizieren, fordert der Server den Client auf, das Passwort **i+1** zu senden. Der Client sendet das entsprechende Passwort. Auf dem Server ist das zuletzt verbrauchte Passwort p_i zusammen mit seinem Index **i** gespeichert. Der Server kennt das Geheimnis **s** nicht, jedoch das zuletzt gültige und das jetzt übermittelte Passwort p_i. Auf diesem, vom Client gelieferten Passwort, wendet der Server die Einwegfunktion genau einmal an. Damit errechnet der Server das ihm bekannte Passwort p_i.

$$p_i = f_{N-i} = f\,(f_{N-(i+1)}\,(s)) = f\,(p_{i+1})$$

Entspricht das generierte Passwort nicht der in der Systempasswortdatei gespeicherten Kopie auf dem Server, ist die Anfrage gescheitert. Stimmen die Passwörter überein, speichert der Server p_{i+1} und den dazugehörigen Index **i+1** (Haller 1994, S. 3–4).

2.3.2.3 Challenge-Response-Verfahren

Eine Verallgemeinerung der Passwortabfrage ist das Challenge-Response-Verfahren. Dem zu authentifizierenden Subjekt werden dabei eine bzw. mehrere Fragen gestellt, die zu beantworten sind. Dieses Challenge-Response-Verfahren wird häufig im Zusammenhang mit Chipkarten oder Authentifizierungsprotokollen verwendet. Dieses Verfahren kann nicht nur einseitig angewendet werden, sondern auch beidseitig, so dass sich beide Beteiligte authentifizieren müssen. Die Sicherheit dieses Verfahrens hängt von der Güte des Zufallszahlengenerators und vom Verschlüsselungsalgorithmus (siehe Kapitel 2.3.1) ab (Eckert 2012, S. 481–485; Rankl und Effing 2002, S. 221–223; Eren und Detken 2006, S. 178f).

2.3.3 Authentifizierung mittels Besitz

Neben der Authentifizierung mittels Wissen gibt es die Möglichkeit der Authentifizierung durch Besitz z. B. Chipkarten. Chipkarten sind Identifikationskarten, in deren Kartenkörper sich eine integrierte Schaltung befindet, die über Elemente zur Datenübertragung, zum Speichern und zur Verarbeitung von Daten verfügt. Es ist möglich, geheime Daten, die Schlüssel, auf eine Chipkarte zu laden, die niemals mehr von außen gelesen oder manipuliert werden können. Zusammen mit der Fähigkeit mit Verschlüsselungsalgorithmen zu rechnen, ermöglicht die Chipkarte die Rea-

lisierung eines handlichen Sicherheitsmoduls. Chipkarten werden durch die Unterschiedlichkeit in Funktionalität und Preis in Speicherkarten und Mikroprozessorkarten unterteilt (Eckert 2012, S. 533–535; Rankl und Effing 2002, S. 20).

Anwendungen für Mikroprozessorkarten sind u. a. Mobilfunknetzkarten, Ausweiskarten, Zugangskontrolle, Zugriffskontrolle auf Rechner, geschützte Datenspeicher oder die elektronische Unterschrift. Vorteile der Mikroprozessorkarte sind:

- Sie besitzen eine höhere Speicherkapazität als Magnetkarten,
- sie speichern sicher geheime Daten,
- sie haben die Fähigkeit, mit Kryptoalgorithmen rechnen zu können,
- sie besitzen eine höhere Lebensdauer und Zuverlässigkeit als Magnetkarten (Rankl und Effing 2002, S. 20f).[18]

Neben dem Besitz der Speicherkarten ist zur Authentifizierung zusätzlich ein gemeinsames Geheimnis notwendig. Bei einfachen PIN-Verfahren wird, wie bei einer Passwortauthentifizierung, das Geheimnis unverschlüsselt zur Karte übertragen. Sicherer ist das Challenge-Response-Verfahren das im Bereich der Mikroprozessorkarten Anwendung findet (Rankl und Effing 2002, S. 219–220).

Die Authentifizierung bei Mikroprozessorkarten kann, wie **Abb. 2-6** zeigt, folgendermaßen gegliedert werden: Wird die Authentizität nur eines Kommunikationspartners sichergestellt, wird von einseitiger Authentifizierung gesprochen. Bei gegenseitiger Authentifizierung werden beide Beteiligte z. B. Sender und Empfänger identifiziert. Als kryptografische Algorithmen im Bereich der Chipkarte kommen symmetrische und asymmetrische zum Einsatz (Rankl und Effing 2002, S. 220).

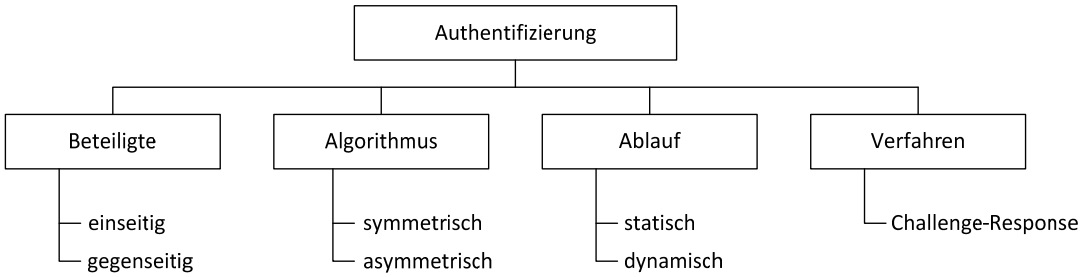

Abb. 2-6 Klassifikationsbaum im Umfeld der Authentifizierung mit Chipkarten nach (Rankl und Effing 2002, S. 220)

[18] Für weitergehende Informationen über allgemeinen Aufbau und Funktionsweise, Anwendungen, physikalische und elektrische Eigenschaften wird auf Rankl und Effing (2002); Rankl (2006); Schier (1999, S. 63–101) verwiesen.

Bei statischen Authentifizierungsverfahren werden immer die gleichen Daten genutzt. Dynamische Verfahren hingegen verwenden für jede Authentifizierung unterschiedliche Datengrundlagen, so dass diese vor einem Angriff mit Einspielen von früher aufgezeichneten Daten besser geschützt sind (Rankl und Effing 2002, S. 219).

Als Authentifizierungsverfahren kommt das Challenge-Response-Verfahren zum Einsatz. Die gegenseitige Authentifizierung des Challenge-Response-Verfahren ist in ISO 9798-2 geregelt.

Damit nicht alle Karten mit den gleichen geheimen Schlüsseln arbeiten, wird in der Praxis ein kartenindividueller Schlüssel verwendet. Bei einem Chipkartenlesegerät und einem Arbeitsplatzrechner ist die Gefahr des Abhörens des Datenverkehrs zur Authentifizierung relativ gering. Im Gegensatz dazu ist die Gefahr höher, wenn eine Authentifizierung zwischen RFID-Karten und Lesegeräten erfolgt (Rankl und Effing 2002, S. 222). Da bei kontaktlosen Chipkarten kein physischer Kontakt zwischen Terminal und Karte besteht, muss die Technik um die Kommunikation zwischen Karte und Lesegerät erweitert werden. Die Daten werden nur auf Anforderung hin übermittelt, erst nachdem ein Taktsignal übertragen wurde, findet der eigentliche Datenaustausch statt (Finkenzeller 2006, S. 7–8).

Für die Beschreibung einer Authentifizierung im Chipkartenbereich wird die Authentifizierung einer RFID-Karte mit einem Lesegerät herausgegriffen. **Abb. 2-7** veranschaulicht den Ablauf eines Challenge-Response-Verfahren einer in RFID-Systemen verwendeten gegenseitigen symmetrischen Authentifizierung, wobei von den technischen Einzelteilen einer RFID-Karte abstrahiert wird. [19]

Eine RFID-Karte kommt in den Lesebereich eines Lesegerätes. Das Lesegerät beginnt die gegenseitige Authentifizierung indem es das Kommando „GET_CHALLENGE“ an die Karte sendet. Die Karte erzeugt daraufhin eine Zufallszahl $\mathbf{R_A}$ als „Challenge“ und sendet diese an das Lesegerät zurück. Das Lesegerät erzeugt ebenfalls eine Zufallszahl $\mathbf{R_B}$. „Unter Verwendung des gemeinsamen geheimen Schlüssels $\mathbf{K}$ und eines gemeinsamen Schlüsselalgorithmus $\mathbf{e_K}$ berechnet das Lesegerät einen verschlüsselten Datenblock (Token1), welcher beide Zufallszahlen und zusätzliche Steuerdaten“ (Finkenzeller 2006, S. 253) beinhaltet. Dieser Token1

[19] Für ein vertiefendes Literaturstudium über Aufbau und Funktionsweise der RFID-Karten sowie technische Lösungen, Anwendungen und Probleme wird auf Finkenzeller (2006); Franke und Dangelmaier (2006); Rankl und Effing (2002) verwiesen.

wird dann als Antwort (Response) an die Karte geschickt (Finkenzeller 2006, S. 253).

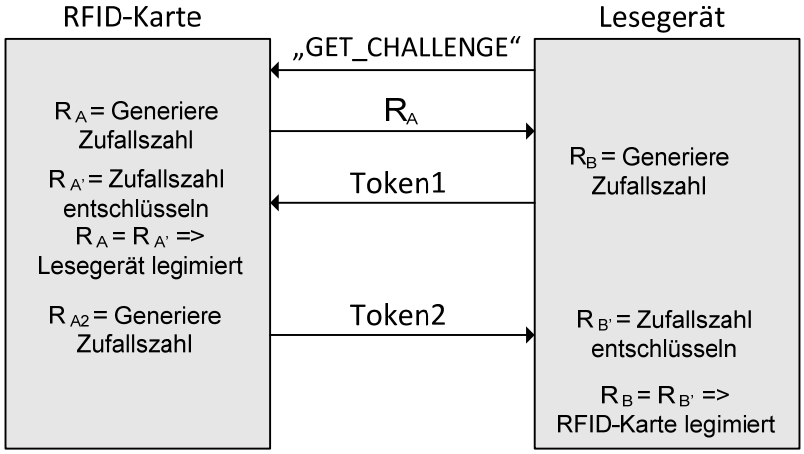

Abb. 2-7 Gegenseitige symmetrische Authentifizierung nach (Finkenzeller 2006, S. 253)

In der Karte wird nun Token1 mittels einer Umkehrfunktion des Verschlüsselungsalgorithmus entschlüsselt und die, im Klartext vorliegende, Zufallszahl $\mathbf{R_{A'}}$ auf Übereinstimmung mit der gesendeten Zufallszahl $\mathbf{R_A}$ geprüft. Stimmen beide Zahlen überein, kennen beide dasselbe Geheimnis und das Lesegerät ist gegenüber der Karte authentifiziert. In der Karte wird nun eine neue Zufallszahl erzeugt. Zusammen mit der verschlüsselten Zufallszahl $\mathbf{R_B}$ und den Steuerelementen wird ein neuer Datenblock (Token2) generiert und an das Lesegerät geschickt. Stimmen $\mathbf{R_B}$ und $\mathbf{R_{B'}}$ überein, so hat sich auch die Karte gegenüber dem Lesegerät legitimiert (Finkenzeller 2006, S. 253).

Das Challenge-Response-Verfahren in einer gegenseitigen symmetrischen Authentifizierung hat folgende Vorteile:

- Der geheime Schlüssel wird nie übertragen, sondern lediglich verschlüsselte Zufallszahlen.
- Es werden immer zwei Zufallszahlen gleichzeitig verschlüsselt. Dadurch kann durch eine Rücktransformation von Token1 und dem Wissen von $\mathbf{R_A}$ nicht auf das gemeinsame Geheimnis geschlossen werden.
- Es können beliebige Algorithmen für die Verschlüsselung der Token verwendet werden.
- Durch die strikte Verwendung von Zufallszahlen aus zwei unabhängigen Quellen bleibt das Aufzeichnen und spätere Wiedervorspielen einer Authentifizierungssequenz ohne Erfolg.

- Aus den erzeugten Zufallszahlen könnte in einem weiteren Schritt ein zufälliger Schlüssel erzeugt werden, um den darauf folgenden Datenverkehr zu verschlüsseln (Finkenzeller 2006, S. 254).

2.3.4 Authentifizierung mittels biometrischer Verfahren

Neben der Authentifizierung mittels Wissen oder Besitz wird nun die dritte Methode mit Hilfe biometrischer Verfahren vorgestellt. Allgemein ist Biometrie die „Lehre von der Zählung und [Körper-]messung an Lebewesen“ (Wermke et al. 2006, S. 256). Im Bereich von Authentifizierungssystemen ist Biometrie der Oberbegriff für alle Verfahren, die Personen durch den Vergleich von unverwechselbaren, individuellen Körpermerkmalen identifizieren. ISO SC37 definiert Biometrics als „automated recognition of individuals based on their behavioural and biological characteristics“ (Bundesamt für Sicherheit in der Informationstechnik (BSI) 2011, S. Glossar 2). Wobei in der Definition „individuals“ im Sinne der Norm auf Menschen begrenzt sind. Die Forschung und Entwicklung von Authentifizierung durch Biometrie befasst sich mit zwei Gebieten, mit körperlichen Merkmalen oder mit typischen Bewegungsmustern. In der Praxis werden als Körpermerkmal zur Identifizierung z. B. Fingerabdruck, Handabdruck, Gesicht oder Augennetzhaut verwendet. Authentifizierung durch typische Bewegungsmuster finden sich bei Sprachidentifizierung, Unterschriftenerkennung oder Tippverhalten des Tastenanschlags (Bundesamt für Sicherheit in der Informationstechnik (BSI) 2011, S. 2; Finkenzeller 2006, S. 4; Peacock et al. 2004; Bakdi 2007).[20]

Biometrische Techniken arbeiten nach folgendem System: Zunächst muss ein Analysesystem Referenzwerte erzeugen. Die zu analysierende biometrische Eigenschaft wird mittels Sensoren erfasst und anschließend digitalisiert. Aus den Referenzwerten werden charakteristische Eigenschaften extrahiert und als personenbezogene Referenzdatensätze auf dem biometrischen Gerät selbst, auf einer personenbezogenen Chipkarte oder in einem anderen Speichermedium abgelegt. Bei einer Authentifizierung wird das erforderliche biometrische Merkmal erhoben, ebenfalls digital transformiert und dann mit den gespeicherten Daten verglichen (Eckert 2012, S. 494–496).

[20] Eine Zusammenfassung über den Stand der Forschung in Bezug auf Authentifizierung durch den Tastenanschlag findet sich in Peacock et al. (2004).

„Die Hauptschwierigkeit biometrischer Verfahren besteht darin, korrekt zu entscheiden, ob aktuelle Werte mit den Referenzwerten übereinstimmen“ (Eckert 2012, S. 494). Bei Authentifizierungsverfahren mittels Wissen (siehe Kapitel 2.3.2) oder Besitz (siehe Kapitel 2.3.3) ist eine Feststellung der Gleichheit zu 100% möglich. Dies ist bei biometrischen Verfahren nicht so einfach möglich, da sich die Eigenschaften einer Person über die Zeit verändern können. Deshalb werden beim Überprüfen der beiden Werte Toleranzschwellen festgelegt. Mittels Korrelationstests sind die Abweichungen von den Referenzwerten zu bestimmen und dann zu prüfen, ob sich diese noch innerhalb der Schwellen bewegen. Bei diesem Abgleich können zwei Typen von Fehlern auftreten:

- Ein berechtigtes Subjekt wird abgewiesen (1).
- Ein unberechtigtes Subjekt wird authentifiziert und akzeptiert (2).

Tritt ein Fehler der ersten Art auf, sind die Kontrollen zu streng, was zu Akzeptanzproblemen seitens der Nutzer führt. Dies wird als tragbar erachtet, wenn dadurch Fehler der zweiten Art mit hoher Wahrscheinlichkeit vermieden werden (Eckert 2012, S. 495–496).

Für die Bewertung der Güte werden die statistisch ermittelte Falschakzeptanzrate (FAR) verwendet, die sich auf den Fehler der ersten Art und die Falschrückweisungsrate (FRR), die sich auf einen Fehler der zweiten Art bezieht. FAR und FRR sind voneinander abhängig, wird FAR minimiert, erhöht sich FRR und umgekehrt. Deshalb wird als zusätzliche Rate die Gleichfehlerrate (EER), der Schnittpunkt der beiden Kurven aus FRR und FAR angegeben. Ziel muss es sein, bei gegebenen FRR eine möglichst kleine FAR zu erhalten. Eine Aussage über die Güte eines biometrischen Verfahrens kann nur getroffen werden, wenn sowohl FRR, FAR und EER angegeben werden. Rahmenbedingungen für die Beurteilung von biometrischen Verfahren sind in der ISO/IEC 29794 festgelegt worden (Eckert 2012, S. 495f).

Nach der Einführung in die verschiedenen Methoden der Authentifizierung mittels Wissen, Besitz oder biometrischer Verfahren wird nun auf die Möglichkeit eingegangen, wie es einem Subjekt ermöglicht wird, durch nur eine einzige Authentifizierung Zugang zu seinen benötigten Anwendungssystemen zu erhalten.

2.3.5 Single-Sign-on im Bereich der Authentifizierung

Durch die verbreitete Nutzung von unterschiedlichen Anwendungssystemen innerhalb des betrieblichen Informationssystems ist es durchaus üblich, dass ein Subjekt viele Zugangsdaten besitzt. Um hier Abhilfe zu schaffen, wurde die Idee entwickelt, dass Nutzer nur ein einziges Authentifizierungsmerkmal besitzen, das für unterschiedliche Anwendungen und Ressourcen verwendet werden kann und keine erneute Authentifizierung erfordert. Bei einer einmal zu Beginn der Sitzung erforderlichen Authentifizierung oder Anmeldung wird von einem "Single-Sign-On" gesprochen. „Single-Sign-On bedeutet, dass das Authentifizierungsverfahren anderen nachfolgend gestarteten Anwendungen Zugriff auf die bestehende Authentifizierungssitzung erlauben muss. Diese müssen zusätzlich in der Lage sein, die Validität der Authentifizierungssitzung ohne weitere Eingaben des Benutzers zu überprüfen“ (Rieger 2007, S. 12). Ein Subjekt braucht damit nur einmal sein Kennwort eingeben und kann anschließend alle Anwendungen und Dienste ohne eine weitere Authentifizierung nutzen (Rieger 2007, S. 12).

In diesem Bereich existieren verschiedene Standards, weshalb ein Single-Sign-On in heterogenen Informationssystemen mit aktuellen Authentifizierungsverfahren und -systemen oftmals nicht möglich ist. „Häufig wird daher bereits lediglich von einer Reduzierung der erforderlichen Authentifizierung für unterschiedliche Applikationen und Ressourcen seitens der Benutzer als "Reduced Sign-On" gesprochen“ (Rieger 2007, S. 12).

2.3.6 Zusammenfassung der Authentifizierung

Zu Beginn jeder Sitzung an einem Arbeitsplatz muss die Authentizität des Subjektes geklärt werden. Für die Überprüfung der Authentizität stehen die folgenden drei Merkmale zur Verfügung. Authentifizierung mittels Wissen, Besitzes oder biometrischer Verfahren. Damit ein Subjekt sich nicht an jedem Anwendungssystem neu anmelden muss, wird an Single-Sign-On-Verfahren gearbeitet, was aber ein einheitliches Identitätsmanagement innerhalb von Organisationen erfordert.

Die Authentifizierung bildet eine notwendige Voraussetzung für eine durchzuführende Zugriffskontrolle. Nur mit einer erfolgreichen Authentifizierung und damit festgestellten Identität kann auch die Autorisierung sichergestellt werden. Die beiden Stufen Authentifizierung und Zugriffskontrolle erfolgen „gerade in heterogenen Um-

gebungen stark differenziert und sind meist nicht untereinander abgestimmt“ (Herwig und Schlabitz 2004, S. 290).

2.4 Zugriffskontrolle

Das folgende Kapitel beschreibt die Grundlagen der Zugriffskontrolle und die Zusammenhänge der Zugriffskontrollstrategie, dem Zugriffskontrollmodell und Zugriffskontrollmechanismus bis hin zum Zugriffskontrollsystem. Dieses stellt für ein Subjekt fest, welche Zugriffe ihm erlaubt sind. Zugriffskontrolle beschäftigt sich im Allgemeinen mit Zugriffen zu physikalischen Systemen, Systemsoftware, AwS oder Daten (Summers 1984, S. 311). Ein Zugriffskontrollsystem umfasst, wie in **Abb. 2-8** dargestellt, die Grundfunktionen Rechteverwaltung, Rechteprüfung und Protokollierung.

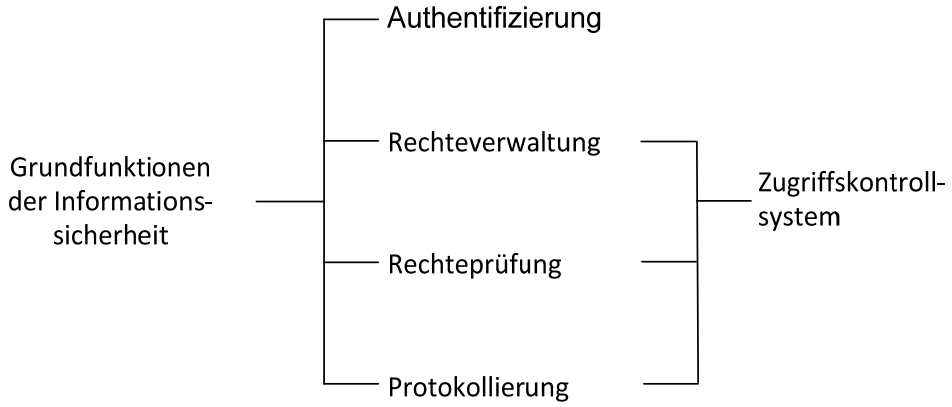

Abb. 2-8 Grundfunktionen der Informationssicherheit (Pohl 2004, S. 682)

Die Rechteverwaltung definiert und speichert alle notwendigen Informationen zu Subjekten und zu überprüfenden Zugriffsrechten, um eine Rechteprüfung durchführen zu können. Die Rechteprüfung kontrolliert vor einem Zugriff, ob ein Zugriffsrecht gewährt werden kann und die Protokollierung dokumentiert erfolgreiche und nicht erfolgreiche Zugriffsversuche. Diese drei Grundfunktionen gewährleisten zusammen mit der Authentifizierung die Erfüllung der Sachziele der Informationssicherheit (siehe Kapitel 2.1).

In der Literatur findet sich im Kontext der Zugriffskontrolle sowohl der Begriff Zugriffskontrolle als auch Autorisierung. Im folgenden werden die beiden Begriffe gegenüber gestellt.

Ausgewählte Beschreibungen für **Zugriffskontrolle**:

- „Die Abwehr von unautorisierten Zugriffen erfordert eine Kontrolle der Zugriffe“ (Eckert 2012, S. 220). Mit der Grundfunktion „der Rechteprüfung ist festzulegen, wann, d. h. bei welchen Aktionen eine Überprüfung stattfinden soll und welche Kontrollen jeweils durchzuführen sind“ (Eckert 2012, S. 220). „Von der Zugriffskontrolle wird gefordert, dass sie jeden Zugriffsversuch kontrolliert und dass diese Kontrolle nicht umgangen werden kann“ (Eckert 2012, S. 628).
- „Die Zugriffskontrolle prüft, ob ein Anwender, der die Authentifizierung passiert hat, berechtigt ist, mit bestimmten Daten zu arbeiten“ (Bill und Zehner 2001).

Ausgewählte Beschreibungen für **Autorisierung**:

- „Unter Autorisierung wird üblicherweise der Vorgang verstanden, bei der einem Subjekt […] Rechte für den Zugriff auf Objekte verliehen werden.“ (Kruth 2001, S. 124).
- Bei der Autorisierung eines Subjektes wird festgelegt, welche Funktionen es ausführen und auf welche Objekte im Rechner es in welcher Weise zugreifen darf (Weck 1984, S. 166).
- „Für den Zugriff auf die zu schützende Information bzw. auf die sie repräsentierenden Daten“ eines Informationssystems „sind Zugriffsrechte festzulegen und an die Subjekte zu vergeben. Besitzt ein Subjekt die Berechtigung zum Zugriff auf eine Information bzw. auf ein Datenobjekt“ so ist „das Subjekt zu diesem Zugriff autorisiert“ (Eckert 2012, S. 5).

Zusammenfassend lassen sich die beiden Begriffe folgendermaßen abgrenzen:

- **Zugriffskontrolle** beschreibt allgemein den Prozess der Kontrolle eines Zugriffs und wird im engeren Sinn synonym zur Grundfunktion Rechteprüfung verwendet.
- **Autorisierung** bezeichnet zum einen „vorgangsorientiert den initialen Zuweisungsprozess der Rechte an einen Benutzer nach dessen Authentisierung am System. Zum anderen bezeichnet er ergebnisorientiert das Resultat des Prüfprozesses, der beim Zugriff auf ein geschütztes Objekt durchlaufen wird“ (Reeg 2012, S. 135).

Um eine Zugriffskontrolle durchführen zu können, müssen durch eine Rechteverwaltung die notwendigen Informationen hinterlegt werden. Deshalb umfasst in dieser Arbeit der Begriff Zugriffskontrolle die Rechteverwaltung, Rechteprüfung und die Protokollierung. Nach dieser begrifflichen Klärung wird anschließend ausgehend von der Zugriffskontrollstrategie auf die Entitäten und Anforderungen für Zugriffskontrollmodelle, die die Grundlage von Zugriffskontrollsystemen bilden, sowie der Implementierung der Zugriffskontrolle, dem Zugriffskontrollmechanismus eingegangen.

2.4.1 Zugriffskontrollstrategie

Ausgangspunkt jedes Zugriffskontrollsystems bildet die Sicherheitsstrategie (siehe Kapitel 2.1.3) (engl. *Security Policy*). Unter Sicherheitsstrategie können allgemein die Festlegung strategischer Ziele und Verfahren zur planmäßigen Herstellung, Überwachung, Erhaltung und Weiterentwicklung der Informationssicherheit verstanden werden (Reeg 2012, S. 75). Die Sicherheitsstrategie formuliert „Schutzziele und allgemeine Sicherheitsmaßnahmen im Sinne offizieller Vorgaben eines Unternehmens oder einer“ (Bundesamt für Sicherheit in der Informationstechnik (BSI) 2011, S. Glossar 11) Verwaltung und beinhaltet Regeln und Verhaltensrichtlinien, die definieren, wie eine Organisation ihre Informationen sicher verwaltet (Eckert 2012, S. 34–35).

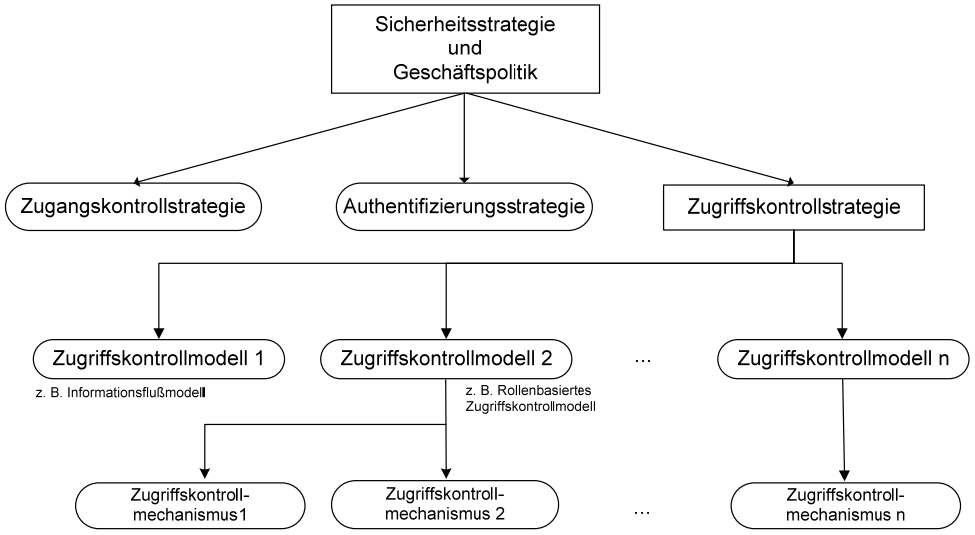

Abb. 2-9 Von der Sicherheitsstrategie und Geschäftspolitik zum Zugriffsmechanismus nach (Seufert 2001, S. 30)

Aus dem Unternehmensplan wird die allgemeine Sicherheitsstrategie eines Unternehmens entwickelt. Aus der Sicherheitsstrategie heraus werden die Zugangskon-

trollstrategie, die Authentifizierungsstrategie und die Zugriffskontrollstrategie abgeleitet. **Abb. 2-9** zeigt die Zusammenhänge zwischen Sicherheitsstrategie bis hin zum Zugriffskontrollmechanismus.

Um anhand der Sicherheitsstrategie und der Geschäftspolitik ein Zugriffskontrollsystem zu entwickeln, wird dieser Vorgang in drei Abstraktionsebenen unterteilt:

- Zugriffskontrollstrategie (access control policies)
- Zugriffskontrollmodell (access control models)
- Zugriffskontrollmechanismen (access control mechanisms)

Die Zugriffskontrollstrategie definiert auf abstrakter Ebene die Zielsetzung, Vorgehensweise und Entscheidungsverfahren bei der Umsetzung der Zugriffskontrolle (Seufert 2001, S. 30). Die Zugriffskontrollstrategie sollte nicht aus Sicht der technischen Sicherheitskonzepte beschrieben werden, sondern die Sicherheitsstrategie des Unternehmens reflektieren. Es können drei Zugriffskontrollstrategien unterschieden werden (Eckert 2012, S. 262–263):

- benutzerbestimmte Zugriffskontrollstrategie,
- systemweite Zugriffskontrollstrategie,
- rollenbasierte Zugriffskontrollstrategie.

Bei einer benutzerbestimmten Zugriffskontrollstrategie werden die Verwaltung und Weitergabe der Zugriffsrechte nach dem Eigentümer- oder Besitzerprinzip durchgeführt. Eigentümerprinzip bedeutet, dass der Eigentümer des Objektes für dessen Schutz verantwortlich ist. Bei einer Umsetzung nach dem Besitzerprinzip reicht der Besitz eines Zugriffsrechtes an einem Objekt aus, um ein Zugriffsrecht weiterzugeben. Jeder Nutzer übernimmt für die Objekte, die seiner Kontrolle unterliegen, die Verwaltung der Zugriffskontrollstrategie. Dabei werden objektbezogene Zugriffsrechte gewährt, die dezentral verwaltet werden. Wenn nicht nur der Eigentümer, sondern auch ein Besitzer eines Rechtes, Zugriffsrechte weitergeben darf, kann es zu inkonsistenten und widersprüchlichen Rechtevergaben kommen (Eckert 2012, S. 262–263; NCSC 1985, S. 11–18).

Die systemweite Zugriffskontrollstrategie ermittelt die Erlaubnis des Zugriffes durch systemglobale Regeln. Die Entwicklung der systemweiten Zugriffskontrollstrategie erfolgte für das Militär. Alle Objekte und Subjekte besitzen eine Klassifikation bezüglich einer Sicherheitsstufe, die sich nicht vom Subjekt beeinflussen lässt. Für

jedes Subjekt wird festgelegt, welche Sicherheitsklasse maximal zugänglich ist. Beispielsweise kann bei einer systemweiten Zugriffskontrollstrategie die Einteilung von Objekten und Subjekten in Sicherheitsklassen (öffentlich, geheim, streng geheim) vorgenommen werden. Diese Strategie ist regelbasiert und wird häufig verwendet, wenn das Sachziel Vertraulichkeit besonders wichtig ist (Eckert 2012, S. 35, 263; NCSC 1985, S. 19–39).

Die beiden oben genannten Strategien kommen nicht nur in Reinform vor. Werden beide Strategien eingesetzt, dominieren die systemweiten Regeln. Das heißt der Zugriff wird verweigert, wenn eine systemweite Regel greift, auch wenn eine benutzerbestimmte Regel den Zugriff erlauben würde. Umgekehrt ist es aber möglich, einen durch eine systemweite Regel gewährten Zugriff durch benutzerbestimmte Regeln weiter einzuschränken (Seufert 2001, S. 40).

Neben diesen beiden in (NCSC 1985) besprochenen Strategien wird in (Eckert 2012, S. 35, 263; Samarati und De Capitani dei Vimercati 2001, S. 41) die rollenbasierte Zugriffskontrollstrategie vorgestellt. Rollen ermöglichen die Zugriffskontrolle möglichst nahe an Aufgaben, organisatorischen Verantwortlichkeiten und Strukturen anzulehnen. „Die zu erfüllenden Aufgaben bestimmen“ die Zugriffsrechte „einer Rolle“ (Lau und Gerhardt 1994, S. 66). Für eine Zugriffskontrolle ist die organisatorische Verantwortung relevanter als die Identität eines Nutzers (Samarati und De Capitani dei Vimercati 2001, S. 41). Die Zugriffskontrolle im betriebswirtschaftlichen Umfeld hängt meist weder von Sicherheitsstufen noch vom Eigentümer eines Objektes ab, sondern von der organisatorischen Einordnung des Subjekts. Durch die rollenbasierte Strategie gelingt es die notwendige Flexibilität bei der Zugriffskontrolle im betriebswirtschaftlichen Umfeld zu erhalten, die für eine effiziente Zugriffskontrolle notwendig ist (Samarati und De Capitani dei Vimercati 2001, S. 41).

Unter Zugriffskontrollmechanismus wird die konkrete technische Umsetzung der Zugriffskontrolle verstanden. Dieser Zugriffskontrollmechanismus übersetzt die Forderung eines Subjektes auf Zugriff in Begriffe wie Tabellenzugriff, Zugriff auf Anwendungssoftware oder Zugriff auf Objekte. Zugriffskontrollmechanismen können Hardware, Software, Prozeduren und Kombinationen aus diesen sein. Sie erlauben letztlich einen autorisierten Zugriff. Zugriffskontrollmechanismen implementieren eine Strategie, um sicherzustellen, dass alle Zugriffe in Übereinstimmung mit dieser erfolgen. Eine Zugriffskontrollstrategie kann durch verschiedene Zugriffskontroll-

mechanismen umgesetzt werden. Ein generischer Ansatz wäre, wenn ein und derselbe Mechanismus mehrere Strategien bedienen könnte. Damit könnte die Strategie geändert werden, ohne den implementierten Zugriffskontrollmechanismus zu ändern (Jajodia 1997, S. 31).

Um die große Lücke der Abstraktion zwischen Strategie und Mechanismus zu schließen, wurden als Bindeglied formale Zugriffskontrollmodelle entwickelt. Für alle drei hier genannten Zugriffskontrollstrategien haben sich im Laufe der Zeit verschiedene Zugriffskontrollmodelle mit zahlreichen Varianten herausgebildet. Im Folgenden werden die Grundlagen von Zugriffskontrollmodellen und Anforderungen an diese vorgestellt.

2.4.2 Zugriffskontrollmodelle

„Die Untersuchung komplexer Systeme erfolgt im Allgemeinen nicht direkt durch Eingriff in das System, sondern indirekt anhand eines geeigneten Modells“ (Ferstl und Sinz 2013, S. 22). Dies gilt ebenso für die Untersuchung im Umfeld der Zugriffskontrolle. Ein Zugriffskontrollmodell (access control model) ist ein formales Modell oder ein Beschreibungsrahmen, mit dessen Hilfe eine Zugriffskontrollstrategie spezifiziert werden kann, um diese automatisiert durchführen zu können. Zugriffskontrollmodelle werden auf einer Abstraktionsebene beschrieben, die es erlaubt eine große Vielseitigkeit von Implementierungsmöglichkeiten in unterschiedlichen Computerumgebungen zu ermöglichen. Ein Zugriffskontrollmodell kann eine oder mehrere Zugriffskontrollstrategien unterstützen (Osborn et al. 2000). Ein Zugriffskontrollmodell definiert zum einen die Entitäten für die Rechteverwaltung, zum anderen umfasst es die Definitionen für die Rechteprüfung.

Bekannte Zugriffskontrollmodelle sind: **Zugriffsmatrix-Modell** (Graham und Denning 1971; Harrison et al. 1976; Bell 2005), **Chinese-Wall Modell** (Clark und Wilson 1987), **Verbandsmodell** (Denning 1976) und **rollenbasiertes Zugriffskontrollmodell** (ANSI INCITS 359-2004 2004; Ferraiolo und Kuhn 1992; Sandhu et al. 1996). In Kapitel 3 wird eine ausführliche Beschreibung und Klassifizierung ausgewählter Modelle vorgenommen.

2.4.2.1 Anforderungen für Zugriffskontrollmodelle

Eine Zugriffskontrollstrategie beschreibt eine Menge von Anforderungen an ein spezifisches Zugriffskontrollsystem. Dem gegenüber ist ein Zugriffskontrollmodell eine eingeschränkte Darstellung eines Zugriffskontrollsystems, das von Details abstrahiert, um die spezifische Struktur und das spezifische Verhalten hervorzuheben. Zugriffskontrollmodelle werden oftmals als formale mathematische Modelle beschrieben, damit Beziehungen und Charakteristiken des Zugriffskontrollsystems erkennbar sind (Amoroso 1994, S. 92). Durch eine mathematische Notation ist es möglich, Beweise durchzuführen (Murauer 2001, S. 4). Einen hohen Grad an Sicherheit in einem AwS zu erreichen, hängt von der Sorgfalt im Design und in der Implementierung von Sicherheitskontrollen ab. Folgende Anforderungen sind an ein Zugriffskontrollmodell zu stellen:

- Es soll präzise und eindeutig sein (Gasser 1988, S. 130).
- Es soll einfach, klar strukturiert und abstrakt sein und von Details abstrahieren, damit es leicht zu verstehen ist (Gasser 1988, S. 130).
- Es soll generisch in Bezug auf die Zugriffskontrolle sein. Es behandelt nur Aspekte der Zugriffskontrolle und nicht Funktionalitäten des AwS (Gasser 1988, S. 130).
- Es soll eine klare Abbildung der Zugriffskontrollstrategie sein (Gasser 1988, S. 130).
- Es ist auf Strukturtreue und Verhaltenstreue zwischen der Zugriffskontrollstrategie und dem Zugriffskontrollmodell zu achten (Ferstl und Sinz 2013, S. 22).

Die Forderung nach Genauigkeit kann erfüllt werden, indem die Modellbeschreibung wenn möglich in einer formalen mathematischen Notation erfolgt. Dies ist besonders dann notwendig, wenn ein Zugriffskontrollsystem in der Art eines Referenzmonitors (siehe Kapitel 2.4.3) entwickelt werden soll.

2.4.2.2 Entitäten des Zugriffskontrollmodells

Über die letzten 30 Jahre hat sich eine Begriffsterminologie entwickelt, um Zugriffskontrollsysteme und -modelle zu beschreiben. Folgende Entitäten finden dabei Verwendung:

- Objekt und Operator

- Nutzer, Sitzung (Session) und Subjekt
- Zugriffsrecht bzw. Recht

Objekt, Operator

Information wird als passive Entität bzw. als Datenobjekt repräsentiert. Ein passives Objekt, z. B. eine Datei, beinhaltet oder erhält Informationen. Ein aktives Objekt, z. B. ein Prozess, kann Informationen speichern, verändern und verarbeiten. Der Zugriff auf ein Objekt bedeutet gleichzeitig Zugriff auf die gespeicherten Informationen. Ein Objekt ist die kleinste Einheit, die durch ein Zugriffskontrollsystem geschützt wird (Eckert 2012, S. 4; NCSC 1988; Seufert 2001, S. 27). Ein Objekt kann eine Ressource, wie ein Drucker oder eine Datei, ein komplexes betriebswirtschaftliches, fachliches Objekt wie eine Rechnung, eine Transaktion oder ein Workflow sein, alles was nach außen als ein Objekt erkennbar ist (Gasser 1988, S. 26; NCSC 1988).

> *Im Prüfungsverwaltungssystem ist ein Objekt z. B. ein Datenblatt eines Studierenden mit dessen persönlichen Daten und Studienverlauf. Es umfasst mehrere Datenbank-Tabellen, ist aber nach außen als ein Objekt erkennbar.*

Komponenten eines Systems, die nicht als Objekte für die Rechteprüfung modelliert sind, unterliegen keiner Zugriffskontrolle, deshalb ist, aus der Sicht von Zugriffskontrollmodellen, die Granularität der Objekte eine wichtige Sicherheitseigenschaft. Unter Granularität wird die Größe von Objekten mit unterschiedlichen Schutzbedürfnissen verstanden (Weck 1984, S. 200; Pohl und Weck 1993; Weck 1993, S. 169).

> *Dies kann z. B. das gesamte Datenblatt eines Studierenden oder ein einzelnes Attribut einer Tabelle in der Datenbank (Note einer Prüfung) sein.*

Eine zu große Granularität kann dazu führen, dass unterschiedliche Entitäten in einem Objekt zusammengefasst werden und damit umfassendere Zugriffsrechte vergeben werden als für die Erfüllung einer Aufgabe notwendig sind. Dies verletzt das Prinzip der minimalen Zugriffsrechte (siehe Kapitel 2.4.3). Damit das Prinzip der minimalen Zugriffsrechte eingehalten werden kann, ist es notwendig, eine anwendungsspezifische Granularität der Objekte modellieren zu können (Eckert 2012, S. 260).

Auf Objekte kann über definierte Schnittstellen (Operationen und Methoden) von anderen Objekten oder der Umwelt des Systems zugegriffen werden (Eckert 2012, S. 5). Eine Operation ist ein aktiver Prozess, der auf einem Objekt angewandt wird. Im Weiteren werden diese Schnittstellen Operatoren genannt (Spies 1985, S. 7).

Objekt und Operator hängen im Wesentlichen vom Zielsystem ab, das durch ein Zugriffskontrollsystem geschützt werden soll. Beispielsweise werden innerhalb eines Dateisystems auf Objekten, wie Dateien und Verzeichnisse, Operatoren wie **Schreiben**, **Löschen** oder **Lesen** angewandt. Bei einem Datenbanksystem sind es Tabellen, die mit den Operatoren **Einfügen**, **Löschen** und **Ändern** gepflegt werden. In einem AwS wie z. B. Prüfungsverwaltungssystem sind mögliche Kombinationen z. B. die Operatoren **Ausdrucken** oder **Lesen** auf dem Objekt **Datenblatt**. Wird ein Operator auf einem Objekt ausgeführt, kann dies als Funktion bezeichnet werden. Es wird dabei die Definition einer Funktion von (Ferstl und Sinz 2013, S. 62–64) zu Grunde gelegt.

Nutzer

Zur Umwelt des Systems gehört der Nutzer, der auf Objekte via Operatoren zugreifen möchte. Der Begriff Nutzer bezieht sich auf personelle Aufgabenträger, die über die Mensch-Computer-Schnittstelle mit einem AwS interagieren. Er ist eine aktive Entität. Ein Nutzer kann mehrere Kennungen besitzen, wobei durch die Authentifizierung eine Kennung genau einem Nutzer zugeordnet werden muss. Einem Nutzer ist es nach einer Autorisierung möglich, Operatoren auf Objekte direkt oder indirekt über Programme anzuwenden.

Sitzung, Session

Eine Instanz eines Nutzerdialogs mit einem System wird Sitzung oder Session genannt. Meldet sich ein Nutzer an, eröffnet er eine Sitzung. Eine Sitzung ist ein Computerprozess, der in Vertretung eines Nutzers agiert und zeitlich beschränkt ist. Eine Sitzung endet entweder mit Abmelden des Nutzers oder nach einer festeingestellten Zeit ohne Eingaben des Nutzers. Auch die Zugriffsrechte sollten nur für die Dauer einer Sitzung gültig sein.

Subjekt

Subjekte sind aktive Entitäten. Ein Subjekt kann eine Person, ein Prozess oder ein Programm, also ein personeller oder maschineller Aufgabenträger sein. Ein aktiver

Nutzer, der eine Sitzung eröffnet hat, wird als Subjekt bezeichnet, das eine Reihe von Funktionen auch stellvertretend durch ein Computerprogramm ausführt. Das Subjekt veranlasst, dass Informationen zwischen Objekten fließen oder der Zustand des Systems sich verändert. In Zugriffskontrollmodellen wird in dieser Arbeit ausschließlich von Subjekt gesprochen. Ein Subjekt ist dabei eine Repräsentation eines personellen Aufgabenträgers in einem Zugriffskontrollmodell (Gasser 1988, S. 26; NCSC 1988; Seufert 2001, S. 28).

Da jeder Zugriff mit Operatoren auf Objekte durch ein Zugriffskontrollsystem autorisiert werden muss, spielt neben der Granularität der Objekte auch die Granularität der Subjekte eine Rolle für die Sicherheit des Informationssystems. Sind die Subjekte grobgranular in Nutzergruppen eingeteilt, die entsprechende Zugriffsrechte erhalten, kann der einzelne Nutzer mehr Zugriffsrechte erhalten als für seine Aufgabe erforderlich ist. Es sollte möglich sein, Subjekte beliebig granular zu modellieren, um eine flexible nutzer- und anwendungsspezifische Zugriffskontrolle sicherzustellen (Eckert 2012, S. 260f).

Zugriffsrecht

Über Zugriffsrechte wird geregelt, welches Subjekt bevollmächtigt wird, Anwendungssysteme, fachliche Objekte oder Daten zu nutzen, um seine Aufgaben zu erfüllen. Ein Zugriffsrecht bezieht sich auf ein Objekt und den dazugehörigen Operator. Dies bedeutet, ein Operator auf zwei verschiedene Objekte sind zwei verschiedene Zugriffsrechte, ebenso wie zwei Operatoren auf ein Objekt. Das Zugriffsrecht ist die wichtigste Komponente in der Zugriffskontrolle (Coyne und Davis 2008, S. 69). Zugriffsrechte beziehen sich in traditionellen Ansätzen auf die Ebene des Betriebssystems. Zugriffsrechte, die sich auf eine Anwendungssoftware beziehen, sind meistens komplexer als bei Betriebssystemen (Beresnevichiene 2003, S. 12), da es in der Anwendungssoftware wesentlich mehr Objekt-Operatoren Kombinationen gibt als in Betriebssystemen.

Das Zugriffsrecht zu einem Zeitpunkt **t** besteht aus dem Tupel $\mathbf{z_t = (s, o, op)}$, aus den Zugriffsrechten, die besagen, dass ein Subjekt **s** den Operator **op** auf dem Objekt **o** ausführen darf. Das von (Spies 1985, S. 5–20) beschriebene formale Modell der Zugriffsrechte in einem Rechensystem kann auf das AwS übertragen werden. Für jede mögliche Funktion, die aufgerufen werden soll, muss das Zugriffskontrollsystem

entscheiden, ob ein Subjekt einen Operator auf einem Objekt ausführen darf. Zugriffsrechte sind bei der Zugriffskontrolle eines AwS das Gegenstück zu den Funktionen im AwS und erlauben die Ausführung dieser Funktionen. Besitzt also ein Subjekt die Berechtigung zum Zugriff auf eine Information bzw. Objekt, so bedeutet dies, dass das Subjekt zu diesem Zugriff autorisiert ist (Eckert 2012, S. 5; Seufert 2001, S. 28; Spies 1985, S. 5–7).

Zugriffsrechte können in zwei Arten unterteilt werden: in universelle und objektspezifische Zugriffsrechte. Bei universellen Zugriffsrechten legt allein das Zugriffskontrollmodell die Operatoren fest, unabhängig von dem zu schützenden System. D. h. es werden allgemeine Operatoren definiert, die nicht vom Objekt abhängen, welches aufgerufen wird, z. B. lesen, schreiben. Werden bei der Zugriffskontrolle nur universelle Zugriffsrechte abgeprüft, kann das Subjekt unnötig Zugriff auf zu viele Informationen erhalten und dies kann für eine Verletzung der Datenintegrität ausgenutzt werden. Von objektspezifischen Zugriffsrechten in Zugriffskontrollmodellen wird gesprochen, wenn Zugriffsrechte aus einem festgelegten funktionalen und fachlichen Kontext heraus auf das jeweilige Objekt zugeschnitten sind. Der fachliche Kontext stellt sicher, dass das Objekt nur nach der festgelegten Semantik des Operators und den durchzuführenden Aufgaben des Subjektes geändert wird. Die Implementierung der objektspezifischen Zugriffsrechte muss von einem entsprechenden Zugriffskontrollsystem unterstützt werden (Eckert 2012, S. 261–262).

2.4.3 Konstruktionsprinzipien sicherer Zugriffskontrollsysteme

Schon seit Beginn der 70er Jahre erschienen Publikationen darüber, welche Konstruktionsprinzipien bei der Entwicklung sicherer Zugriffskontrollsysteme zu berücksichtigen sind. Die durch (Saltzer und Schroeder 1975) veröffentlichten allgemeinen Konstruktionsprinzipien sicherer Systeme haben bis heute Gültigkeit (Eckert 2012, S. 188). Neben diesen Konstruktionsprinzipien stellt das Modell des Referenzmonitors Vollständigkeit, Überprüfbarkeit und Isolation in den Mittelpunkt (Anderson 1972, S. 15–27). Die Prinzipien des Referenzmonitors werden um Flexibilität, Skalierbarkeit und Verwaltbarkeit erweitert (Ferraiolo et al. 2003, S. 31–35). Die im Folgenden beschriebenen Konstruktionsprinzipien für Zugriffskontrollsysteme umfassen die Konstruktionsprinzipien von (Saltzer und Schroeder 1975), den Refe-

renzmonitor und dessen Erweiterungen und können in drei Gruppen eingeteilt werden:

- Prinzipien, die auf Ebene des Zugriffskontrollmodells,
- Prinzipien, die im Zugriffskontrollmodell und in der Implementierung,
- Prinzipien, die erst bei der Implementierung

berücksichtigt werden müssen.

Prinzipien, die schon auf Ebene der Zugriffskontrollmodelle zu berücksichtigen sind, werden zur Beurteilung der Zugriffskontrollmodelle in Kapitel 3.3 herangezogen.

Prinzipien, die auf Ebene des Zugriffskontrollmodells einbezogen werden sollten:

- **Aufgabentrennung**: Kritische Aufgaben in einem Unternehmen dürfen nicht von ein und derselben Person ausgeführt werden. Aufgabentrennung (Trennung der Privilegien; im Englischen „Separation of Duty“) (Saltzer und Schroeder 1975, S. 1283) wird immer dort eingesetzt, wo anwendungsspezifische Sicherheitsrichtlinien erforderlich sind. Das Telecom Glossary beschreibt Aufgabentrennung als: „In secure communications, dividing responsibility for sensitive information so that no individual acting alone can compromise the security of the data processing system” (ATIS Telecom Glossary 2007). Aufgabentrennung ist ein Sicherheitsprinzip, um Kontrollstrategien über mehrere Subjekte zu formulieren, die gemeinsam verantwortlich sind, eine Aufgabe vollständig zu erledigen (Simon und Zurko 1997, S. 183). Aufgabentrennung ist eine anwendungsspezifische, keine systemweite Sicherheitsrichtlinie (Gligor et al. 1998, S. 1), die soweit sie das Zugriffskontrollsystem betrifft, bereits auf Modellebene berücksichtigt werden muss. Aufgabentrennung hat zwei Aspekte: eine funktionale Trennung und das 4-Augenprinzip (Anderson 2001, S. 189–190). Bei kritischen Zugriffen auf Objekte sollte das 4-Augenprinzip angewandt werden. Eine Kontrollinstanz darf die zu kontrollierende Information zum Beispiel lesen, aber nicht verändern. So dürfen z. B. ein Buchhalter und ein Controller nicht die dieselben Zugriffsrechte an den gleichen Objekten besitzen[21].

[21] Weiterführende Literatur sowie Beispiele finden sich in Anderson (2001, S. 166–199); Brewer und Nash (1989); Clark und Wilson (1987); Gligor et al. (1998); Nash und Poland (1990); Thomas und Sandhu (1994).

- **Skalierbarkeit**: Die Verwaltung und Überprüfung von Zugriffsrechten soll in Bezug auf die Anzahl der Subjekte und Objekte für alle Anwendungssysteme des gesamten Unternehmens beliebig skalierbar sein. Die Granularität muss beliebig erfolgen können (siehe Kapitel 2.4.2.2) (Ferraiolo et al. 2003, S. 31–35).

Prinzipien, die im Zugriffskontrollmodell und der Implementierung berücksichtigt werden:

- **Überprüfbarkeit:** Das Design des Modells muss so einfach und klein wie nur möglich sein, um eine Überprüfung zu ermöglichen. Dies gilt ebenso für die Implementierung des Referenzmonitors bzw. ISO Access Frameworks (siehe Kapitel 2.4.4), um ihn hinsichtlich der Vollständigkeit des gewährleisteten Sicherheitsbegriffs leicht analysieren und testen zu können (Saltzer und Schroeder 1975, S. 1282).
- **Prinzip der minimalen Zugriffsrechte**: Dieses Prinzip, auch bekannt als „need-to-know–Prinzip“ (Saltzer und Schroeder 1975, S. 1283), „Prinzip der least privileges“ oder „Prinzip der minimalen Privilegien“, fordert, dass jedes Subjekt nur die Zugriffsrechte erhalten darf, „die es zur Erfüllung seiner Aufgaben mindestens benötigt“ (Müller 2005, S. 302). Durch die Beachtung des Prinzips der minimalen Zugriffsrechte soll vermieden werden, dass ein Subjekt die Möglichkeit bekommt, unerlaubte Aktionen, als Ergebnis einer unnötigen Zuordnung von Zugriffsrechten zu einem Subjekt, ausführen zu können. (Gasser 1988, S. 49–50).
- **Flexibilität:** Die Zugriffskontrollstrategie des gesamten Unternehmens soll unterstützt werden.

Prinzipien, die bei der Implementierung beachtet werden:

- **Erlaubnisprinzip:** Die Zugriffsentscheidung basiert auf Erlaubnis und nicht auf Zurückweisung des Zugriffs (Saltzer und Schroeder 1975, S. 1282). Diese konservative Denkweise bedeutet, das vordefinierte Verhalten ist eine Verweigerung des Zugriffs beim Fehlen eines Zugriffsrechtes, d.h. versucht ein Subjekt ein Objekt aufzurufen und wird dafür kein Zugriffsrecht gefunden, wird der Zugriff verweigert. Dies wird als geschlossenes Zugriffskontroll-

system bezeichnet[22]. Im Gegensatz dazu sind in einem offenen System Zugriffe, die nicht explizit verboten sind, erlaubt (Castano et al. 1995, S. 20f).

- **Vollständigkeit:** Jeder Zugriff muss durch das Zugriffskontrollsystem überprüft werden (Saltzer und Schroeder 1975, S. 1283).
- **Offener Entwurf**: Dieses Prinzip besagt, dass die Verfahren und Mechanismen offen gelegt werden müssen, da die Sicherheit eines Systems nicht von der Geheimhaltung der Verfahren abhängig sein darf (Saltzer und Schroeder 1975, S. 1283).
- **Akzeptanz bzw. Verwaltbarkeit**: Das Prinzip fordert, dass das Zugriffskontrollsystem einfach zu bedienen sein muss, so dass der Nutzer den Schutzmechanismus einfach und routinemäßig ausführen kann (Saltzer und Schroeder 1975, S. 1283).
- **Isolation:** Das Zugriffskontrollsystem, das den Zugriff kontrolliert, muss gegenüber unerlaubten Änderungen sicher sein (Anderson 1972, S. 15–27).

2.4.4 Implementierung eines Zugriffskontrollsystems

Seit Beginn der Diskussion über Zugriffskontrolle wird daran gearbeitet, wie eine überprüfbare und geeignete Implementierung der Rechteprüfung gewährleistet werden kann. Das Modell des Referenzmonitors (Anderson 1972) findet in Betriebssystemen bis heute Anwendung. Der Referenzmonitor validiert alle Zugriffe von Subjekten auf Objekte. Das Modell hat in die Norm ISO 10181-3 Eingang gefunden. Diese Norm sieht ebenfalls die Implementierung eines Referenzmonitors für eine Zugriffskontrolle vor, allerdings wird in diesem Framework zusätzlich noch eine zentrale Entscheidungsfunktion vorgeschlagen, die anhand der gespeicherten Daten die Zulässigkeit des Zugriffs ermittelt.

Zugriffskontrollstrategien können Geschäftsprozessregeln und Sicherheitsregeln beinhalten. Für die Implementierung von Zugriffskontrollsystemen ist darauf zu achten, dass Geschäftsprozessregeln und Sicherheitsregeln getrennt werden. Sicherheitsregeln sind vom Zugriffskontrollsystem zu überprüfen. Werden Sicherheitsregeln und Geschäftslogik gemischt, wird das Ergebnis komplexer und es ist schwerer zu verstehen, warum ein Zugriff gewährt bzw. verwehrt wurde. Während ein Zugriffskon-

[22] Eine Definition sowie Vor- und Nachteile von geschlossenen und offenen Zugriffskontrollsystemen beschreibt Castano et al. (1995, S. 20–21).

trollsystem die Sicherheitsregeln einer Organisation umsetzt, werden im Anwendungssystem die Geschäftsprozessregeln implementiert (Coyne und Davis 2008, S. 69–70).

2.4.4.1 Modell des Referenzmonitors

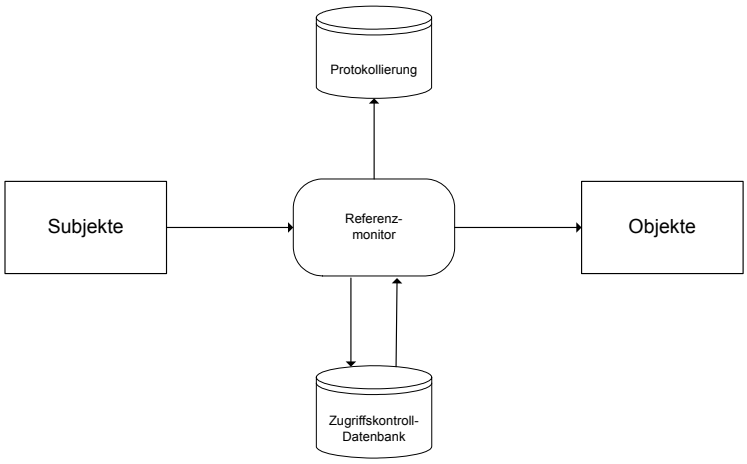

Abb. 2-10 Referenzmonitor (Ferraiolo et al. 2003, S. 32)

Das abstrakte Modell des Referenzmonitors stellt die autorisierte Zugriffsbeziehung zwischen einem Subjekt und Objekt sicher und protokolliert die Zugriffe. Das Modell des Referenzmonitors, siehe **Abb. 2-10**, diktiert weder die Zugriffskontrollstrategie noch einen bestimmten Zugriffskontrollmechanismus. Der Referenzmonitor kann ein Teil der Hardware oder der Software eines Betriebssystems sein und ist für die Durchsetzung der Zugriffskontrolle laut Zugriffskontrollstrategie verantwortlich (Anderson 1972, S. 15–27; NCSC 1985, S. 65).

Neben dem Prinzip der Vollständigkeit und Überprüfbarkeit wird bei der Konstruktion von Betriebssystemen mit dem Modell des Referenzmonitors das Prinzip der Isolation umgesetzt. Alle sicherheitsrelevanten Dienste und Programme sind im Sicherheitskern des Referenzmonitors zusammengefasst und von den übrigen Systemkomponenten getrennt. Ein solcher Ansatz reduziert den Aufwand für die Verifikation (Weck 1984, S. 192). Die Überprüfbarkeit kann durch die Verwendung von Softwareprinzipien wie Modularisierung, abstrakte Spezifikation und Information hiding erreicht werden und wird versucht durch Codeinspektion, Testen, formale mathematische Spezifikationen und Verifikationen sicherzustellen (Gasser 1988, S. 168).

Die Anforderungen des Referenzmonitors sind notwendig, aber nicht ausreichend für die Umsetzung sicherer Zugriffskontrollsysteme. Ein Zugriffskontrollsystem sollte

neben der Sicherheit, die der Referenzmonitor zur Verfügung stellt, auch noch Flexibilität, Verwaltbarkeit und Skalierbarkeit sicherstellen (Ferraiolo et al. 2003, S. 31–35). Das Modell des Referenzmonitors wurde mit dem Fokus auf Betriebssysteme entwickelt. Die Erfahrung zeigt, dass Betriebssysteme, deren Schutzmaßnahmen entsprechend dem Konzept des Referenzmonitors strukturiert sind, im allgemeinem ein sehr hohes Maß an Sicherheit gegen Penetrationsversuche aufweisen (Weck 1993, S. 150). Die für Betriebssysteme implementierten Prinzipien können auf die Zugriffskontrolle in Anwendungssoftware übertragen werden. Die Idee des Referenzmonitors bildet die Grundlage für das ISO Access Control Framework, das nachfolgend vorgestellt wird.

2.4.4.2 ISO Access Control Framework

Als Basis für die Implementierung der Zugriffskontrolle wird das Framework des dritten Teils des Standards ISO 10181-3 herangezogen. Die Sicherheitsdomäne besteht aus einer Sicherheitsstrategie, einem zuständigen Sicherheitsadministrator und einer Menge von Objekten und Operatoren. Die Sicherheitsstrategie bestimmt unter welchen Bedingungen welche Operatoren auf welchen Objekten ausgeführt werden dürfen (Biltzinger und Bunz 2004, S. 30–32).

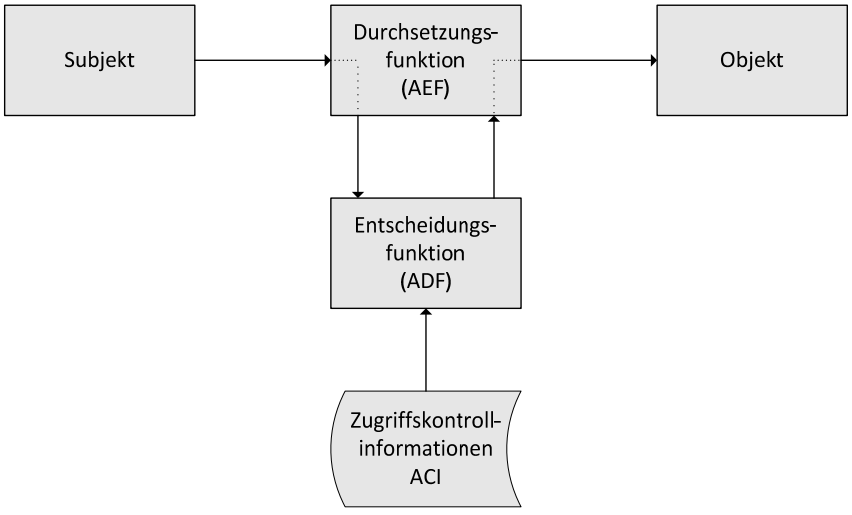

Abb. 2-11 Autorisierungsmodell ISO 10181-3 nach (Biltzinger und Bunz 2004, S. 31)

Der Standard definiert folgende Komponenten, wie **Abb. 2-11** zeigt (in spitzen Klammern werden die Begriffe aus Kapitel 2.4.2.2 hinzugefügt):

- Initiator <Subjekt>: Eine Entität, z. B. eine Person oder eine Anwendung, die den Zugriff auf eine Ressource <Objekt> verlangt.
- Target <Objekt>: Eine IT Ressource, auf die der Zugriff ersucht wird.

- Access Enforcement Funktion (AEF) <Referenzmonitor>: Funktion, die bei jedem Zugriff aufgerufen wird, um sicherzustellen, dass nur berechtigte Zugriffe erlaubt werden.
- Access Decision Funktion (ADF): Funktion, die entscheidet, ob eine Zugriffsanfrage berechtigt ist oder nicht (Zugriffsentscheidungsfunktion).
- Access Control Information (ACI) <Rechteverwaltung>: Jede Information, die für die Zugriffskontrolle relevant ist, muss gespeichert werden.

Ein Subjekt möchte Zugriff auf ein Objekt erhalten, dazu wird die Anfrage nach dem Vollständigkeitsprinzip an die Durchsetzungsfunktion (AEF) weitergeleitet. Diese übergibt an die Entscheidungsfunktion (ADF) die notwendigen Parameter, und liefert, in dem sie alle zur Verfügung stehenden Zugriffskontrollinformationen (ACI) auswertet, das Ergebnis der Entscheidung an die Durchsetzungsfunktion zurück. Bei einer Zurückweisung erhält das Subjekt eine entsprechende Fehlermeldung. Die Entscheidung, ob ein Zugriffsrecht gewährt wird, findet ausschließlich im ADF statt (Biltzinger und Bunz 2004, S. 30–33; Lehmann 2007, S. 14–15). Die Einführung eines solchen Frameworks zur Überprüfung, ob ein Zugriffsrecht besteht, hat großen Einfluss auf die Architektur und Entwicklung des gesamten Zugriffskontrollsystems (Kern et al. 2004, S. 88f). Der Standard bezieht sich allgemein auf Zugriffskontrollsysteme und damit auch auf Zugriffskontrollsysteme für Anwendungssysteme und ist als Weiterentwicklung des Referenzmonitors zu sehen.

2.5 Zusammenfassung

Informationssicherheit insbesondere IT-Sicherheit wird gewährleistet, wenn die Erfüllung der drei Sachziele Vertraulichkeit, Integrität und Verbindlichkeit sichergestellt wird. Das Ziel der Verfügbarkeit muss durch Systemintegrität und Funktionssicherheit des Anwendungssystems erreicht werden. In der Diskussion um die Sachziele der Informationssicherheit wird Verfügbarkeit als eine Voraussetzung für die anderen Ziele gesehen (Pohl und Weck 1993, S. 16).

Die IT-Sicherheit wird durch die Grundfunktionen

- Authentifizierung und
- Zugriffskontrolle (Rechteverwaltung, Rechteprüfung und Protokollierung)

gewährleistet. Die Authentifizierung ist der Zugriffskontrolle vorgelagert und ist für diese eine notwendige Voraussetzung. Der Datenschutz bzw. das BDSG selbst be-

nennt die Grundfunktionen nicht explizit, nimmt aber mit seinen Ausführungsbestimmungen Bezug auf diese.

Für die Umsetzung der Zugriffskontrolle werden drei Abstraktionsebenen gebildet: Zugriffskontrollstrategie, Zugriffskontrollmodell und Zugriffskontrollmechanismus. Nach der Einführung in die Terminologie und Entitäten der Zugriffskontrolle wurden die Konstruktionsprinzipen für sichere Zugriffskontrollsysteme vorgestellt und den entsprechenden Abstraktionsebenen Zugriffskontrollmodell oder Zugriffsmechanismus zugeordnet. Die Entitäten und die Konstruktionsprinzipien dienen als Grundlage für die Beurteilung der einzelnen Zugriffsmodelle, die im nächsten Kapitel vorgestellt werden. Die Zugriffskontrollmodelle sollten die Konstruktionsprinzipien vom Modell her schon unterstützen und dienen als Kriterium zur Auswahl des Zugriffskontrollmodells für die Entwicklung eines Zugriffskontrollsystems.

Ein Zugriffskontrollsystem muss nach dem Access Control Framework nach ISO 10181-3 entwickelt werden. Die Rechteverwaltung sollte durch eine zentrale Administration vorgenommen werden. Die Rechteprüfung muss als Referenzmonitor implementiert werden und muss jeden Zugriff auf die Anwendungssoftware und auf die von extern aufrufbaren Funktionen anhand der Zugriffsrechte des zugreifenden Subjektes überprüfen. Das dritte Sachziel, die Verbindlichkeit, wird durch eine Protokollierung der Zugriffe erreicht.

3 Modelle der Zugriffskontrolle

Seit Anfang der 70er Jahre werden formale Modelle (siehe Kapitel 2.4.2) für die Beschreibung und Untersuchung der Zugriffskontrolle entwickelt. Diese sollen u. a. zeigen „wenn ein Zugriffskontrollmodell im Initialzustand ‚sicher' ist, dann darf es durch keine der Operationen auf dessen Objekte den Zustand ‚sicher' verlassen" (Murauer 2001, S. 4). „The purpose of the so-called 'security model' is to provide a basis for determining whether or not a system is secure, and if not, for detecting its flaws" (Goguen und Meseguer 1982, S. 12). Formale Modelle eignen sich dazu, verbal gegebene Sicherheitsanforderungen präzise zu erfassen, auf Konsistenz zu überprüfen und eine Realisierung vorzubereiten. Allerdings gibt es nicht ein Zugriffskontrollmodell für alle Anwendungsfälle, sondern jedes Anwendungsszenario hat seine eigenen Sicherheitsanforderungen und benötigt unter Umständen ein anderes Zugriffskontrollmodell, um seine Zugriffskontrollstrategie umzusetzen (Kessler 1992, S. 465).

Nachfolgend werden nun ausgewählte Zugriffskontrollmodelle vorgestellt und untersucht. Dazu wird zunächst ein Klassifikationsrahmen erstellt und die Zugriffskontrollmodelle darin eingeordnet, gegenüber gestellt und bewertet.

3.1 Klassifikationsrahmen für die Einordnung von Zugriffskontrollmodellen

„Unter Klassifikation wird ganz allgemein eine Gruppierung oder Einteilung" eines Wissensgebietes „nach einheitlichen methodischen Prinzipien verstanden" (Manecke 2004, S. 127). Sie dient damit als Ordnungsmittel für einen Gegenstandsbereich (Kuhlen et al. 2004, S. 69). Zuerst wurden Zugriffskontrollmodelle nur anhand der Sicherheitsstrategie in systemweite und benutzerbestimmte Zugriffskontrollmodelle eingeteilt. In jüngerer Literatur finden sich auch Klassifikationen mit umfangreicheren Bewertungskriterien. Das nachfolgend entwickelte Klassifikationsschema stützt sich auf die beiden Klassifikationen von (Seufert 2001, S. 37–45) und (Eckert 2001, S. 125–131; Eckert 2012, S. 269–273). Die beiden Klassifikationen sind nicht genau deckungsgleich, verwenden z. T. unterschiedliche Begriffe für denselben Sachverhalt, sind aber nicht unabhängig voneinander, sondern lassen sich ineinander überführen. Nach einer Gegenüberstellung der beiden Klassifikationen wird eine Zusammenführung, Systematisierung und Erweiterung vorgenommen. Daran

schließt sich eine Überprüfung, ob die Klassifikation vollständig, überschneidungsfrei und orthogonal zueinander ist, an.

Die Frage der Verwendbarkeit eines spezifischen Zugriffskontrollmodells für die IT-Sicherheit muss sich „daran orientieren, welche problemspezifischen Anforderungen bestehen und welche Möglichkeiten das jeweilige Zugriffskontrollmodell bietet, diese Anforderungen zu erfassen“ (Eckert 2012, S. 269). Die Zugriffskontrollmodelle müssen im Hinblick auf durch sie beschreibbare Eigenschaften analysiert und entsprechend der Anforderungen beurteilt werden (Eckert 2012, S. 229).

3.1.1 Gegenüberstellung der Klassifikationen

Die Kriterien der beiden Klassifikationen von (Seufert 2001) und (Eckert 2001, S. 125–131; Eckert 2012, S. 269–273) werden nachfolgend gegenübergestellt:

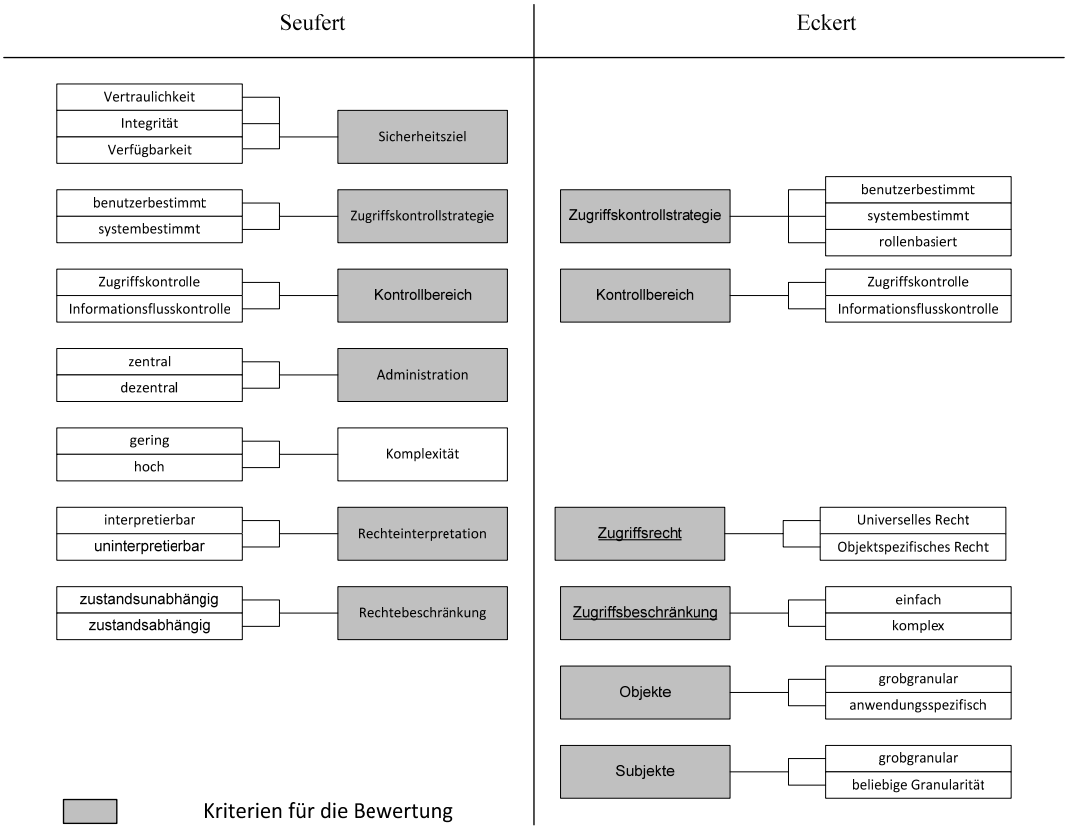

Abb. 3-1 Gegenüberstellung der Klassifikationskriterien von (Seufert 2001, S. 37–45) und (Eckert 2001, S. 125–131; Eckert 2012, S. 269–273)

Kriterien, die sich in **Abb. 3-1** auf einer Horizontalen befinden, sind als Synonyme zu betrachten und beschreiben denselben Sachverhalt. Die grau hinterlegten Kriterien werden zur Klassifikation und Beurteilung der ausgewählten Zugriffskontrollmodelle

herangezogen. Im weiteren Verlauf der Arbeit werden die unterstrichenen Begriffe verwendet.

Sicherheitsziele

Die Sicherheitsziele umfassen die Sachziele der Informationssicherheit (siehe Kapitel 2.1.1). Zugriffskontrollmodelle fokussieren unterschiedliche Schwerpunkte bei den Sachzielen der Informationssicherheit. Das Einsatzgebiet und das verfolgte Sachziel stehen im engen Zusammenhang. Im militärischen Bereich steht das Sachziel Vertraulichkeit an oberster Stelle, im kommerziellen und administrativen Bereich ist die Integrität der Daten am wichtigsten (Clark und Wilson 1987, S. 188). (Seufert 2001, S. 41–42) klassifiziert die Zugriffskontrollmodelle nach den Sachzielen: Integrität, Vertraulichkeit und Verfügbarkeit. Wobei Verfügbarkeit nicht in die Anforderungen übernommen wurde (siehe Kapitel 2.1.1). (Eckert 2001, S. 130–131) definiert vier Klassen, diese gewichten die Sachziele Integrität und Vertraulichkeit und werden orthogonal zu den nachfolgend beschriebenen Kriterien verwendet.

Zugriffskontrollstrategie

Die Zugriffskontrollstrategie definiert die Vorgehensweise und die Zielsetzung bei der Umsetzung des Schutzes von Informationen. Sie beinhaltet die Art der Kontrolle sowie die Definition der Regeln für einen Zugriff. Die Zugriffskontrollstrategie wird bei (Seufert 2001, S. 40) von (NCSC 1985, S. 11–39) übernommen und in systemweite und benutzerbestimmte Zugriffskontrollstrategie eingeteilt. Zusätzlich wurde von (Eckert 2012, S. 262–263) eine rollenbasierte Zugriffskontrollstrategie eingeführt.

Kontrollbereich

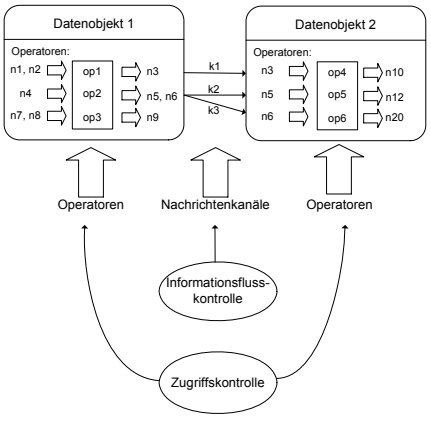

Abb. 3-2 Kontrollbereiche der Zugriffskontrolle (Seufert 2001, S. 42)

Der Kontrollbereich wird in Zugriffskontrolle und Informationsflusskontrolle eingeteilt. **Abb. 3-2** stellt diesen Unterschied grafisch dar. Die Zugriffskontrolle bezieht sich direkt auf einen Zugriff über die Operatoren auf ein Objekt und überprüft vor dem Zugriff die Berechtigung. Die Informationsflusskontrolle kontrolliert hingegen die Nachrichtenkanäle, mit denen die Information zwischen Objekten ausgetauscht werden (Eckert 2012, S. 263; Seufert 2001, S. 41).

Administration

Es wird zwischen einer zentralen und dezentralen Administration unterschieden (Seufert 2001, S. 43).

Komplexität

Dieses Kriterium bezieht sich auf die Komplexität bei der Implementierung und Konfiguration des Zugriffskontrollsystems. Eine geringe Komplexität liegt vor, wenn eine Umsetzung mit einfachen Datenstrukturen möglich ist. Eine hohe Komplexität bedeutet, dass aufwendige Algorithmen und komplexe Datenstrukturen notwendig sind (Seufert 2001, S. 44). Das Kriterium Komplexität stellt bei der Einordnung, Beurteilung und Auswahl eines Zugriffskontrollmodells ein untergeordnetes Kriterium dar. Das Argument, die Administration darf nicht komplex sein bzw. die Implementierung sollte nicht auf komplexen Datenstrukturen basieren, ist kein geeignetes Auswahlkriterium. Es muss zwischen Implementierung und Administration unterschieden werden. Durch eine geeignete Implementierung sollte die Administration einfach und die Zugriffskontrolle nach dem ISO Access Control Framework sicher durchführbar sein.

Rechteinterpretation/Zugriffsrecht

Das Kriterium Rechteinterpretation bzw. Zugriffsrecht beschreibt, ob das Modell Zugriffsrechte an Operatoren bzw. Operatorentypen bindet. Die zwei möglichen Ausprägungen dieses Kriteriums sind interpretierbar bzw. **universell** und uninterpretierbar bzw. **objektspezifisch** (siehe 2.4.2.2). Universelle Zugriffsrechte sind im Modell festgeschriebene allgemeine Operatoren, deshalb ist ein solches Modell nur schwer auf andere Operatorentypen übertragbar. Objektspezifische Zugriffsrechte ermöglichen den Zugriff auf einen funktionalen Kontext des jeweiligen Objektes zu beschränken (Eckert 2012, S. 260; Seufert 2001, S. 43).

Rechtebeschränkung/Zugriffsbeschränkung

Bei der Rechtebeschränkung bzw. Zugriffsbeschränkung werden zwei Ausprägungen, einfache bzw. **zustandsunabhängige** und komplexe bzw. **zustandsabhängige** Zugriffsrechte betrachtet. Bei **zustandsunabhängigen** Zugriffsrechten definiert allein das Zugriffskontrollmodell den Zugriff. Bei **zustandsabhängigen** Zugriffsrechten ist eine Autorisierung zusätzlich vom zeitlichen oder fachlichen Kontext abhängig (Eckert 2012, S. 262; Seufert 2001, S. 44).

Objekte

Die Ausprägungen des Kriteriums **Objekte** (siehe Kapitel 2.4.2.2) sind: feingranular, im Sinne von **anwendungsspezifisch**, oder **grobgranular**. Ist nur eine grobgranulare Einteilung z. B. Datei oder Tabelle möglich, kann das Prinzip der minimalen Zugriffsrechte nicht zuverlässig gesichert werden. Anwendungsspezifisch bedeutet, dass die Granularität unabhängig vom Zugriffskontrollmodell ist und an das zu überprüfende Anwendungssystem angepasst werden kann (Eckert 2012, S. 260–261).

Subjekte

Beim Kriterium **Subjekte** (siehe Kapitel 2.4.2.2) werden die Ausprägungen **grobe** bzw. **beliebige** Granularität festgelegt. Eine grobkörnige Einteilung bedeutet, dass Zugriffsrechte nicht einzelnen Subjekten, sondern z. B. nur Nutzergruppen zugeordnet werden können. Hier besteht ebenfalls die Gefahr, dass das Prinzip der minimalen Zugriffsrechte verletzt wird. Bezieht sich die Rechteverwaltung und Rechteprüfung auf einzelne Subjekte oder Prozesse, wird von beliebiger Granularität der Subjekte gesprochen (Eckert 2012, S. 260–261).

3.1.2 Einordnung in das Modell des Zugriffs

Zusätzlich zu den Kriterien der beiden im vorherigen Abschnitt beschriebenen Klassifikationen werden die beiden Kriterien *Aufgabentrennung* und *Prinzip des minimalen Rechtes* aus den Konstruktionsprinzipien (siehe Kapitel 2.4.3) in die Klassifikation aufgenommen. Um Vollständigkeit, Überschneidungsfreiheit und Orthogonalität überprüfen zu können, werden die Kriterien in das folgende Modell des Zugriffs (siehe **Abb. 3-3**) eingeordnet. Ein Subjekt greift in einem Anwendungssystem auf ein Objekt mit seinen Operatoren zu, um Informationen zu erhalten oder zu verändern.

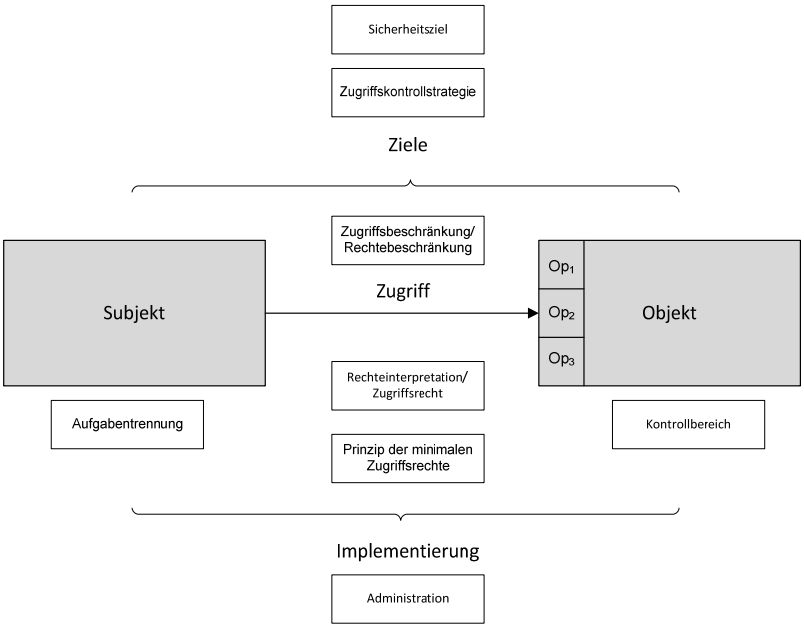

Abb. 3-3 Einordnung der Begriffe für die Klassifizierung der Zugriffskontrolle in das Modell des Zugriffs

Die Sicherheitsziele definieren, wie ein Zugriff überwacht werden soll. Die Implementierung sorgt für die Übertragung des Modells in ein Zugriffskontrollsystem und ist verantwortlich für die Rechteverwaltung, Rechteprüfung und Protokollierung. Die Kriterien Sicherheitsziel und Zugriffskontrollstrategie sind den Zielen der Zugriffskontrolle zuordenbar. Das Kriterium **Objekte** ist dem Objekt zuzuordnen. Das Kriterium **Subjekte** sowie die Aufgabentrennung können dem Bereich des Subjekts zugerechnet werden. In den Bereich des Zugriffs fallen Zugriffs- bzw. Rechtebeschränkung, Zugriffsrecht bzw. Rechteinterpretation sowie das Prinzip der minimalen Zugriffsrechte. Eine geeignete Implementierung des Zugriffskontrollmechanismus muss für eine überschaubare Administration und sichere Rechteprüfung sorgen.

3.1.3 Anforderungen an Zugriffskontrollmodelle

Die in **Abb. 3-4** dargestellten Kriterien werden als Grundlage für die Beurteilung der Zugriffskontrollmodelle am Ende des Kapitels herangezogen. Die Klassifikation setzt sich wie folgt zusammen:

- Die Sicherheitsziele, also die Sachziele für die IT-Sicherheit, sind Vertraulichkeit, Integrität und Verbindlichkeit (siehe Kapitel 2.1.1). Die Verbindlichkeit wird erst durch eine geeignete Implementierung gewährleistet. Jedoch sollte ein Zugriffskontrollmodell die Möglichkeit bieten, die Objekte

und Subjekte so zu modellieren, dass eine ausreichende Protokollierung erreicht wird, anhand derer die Verbindlichkeit überprüft werden kann.

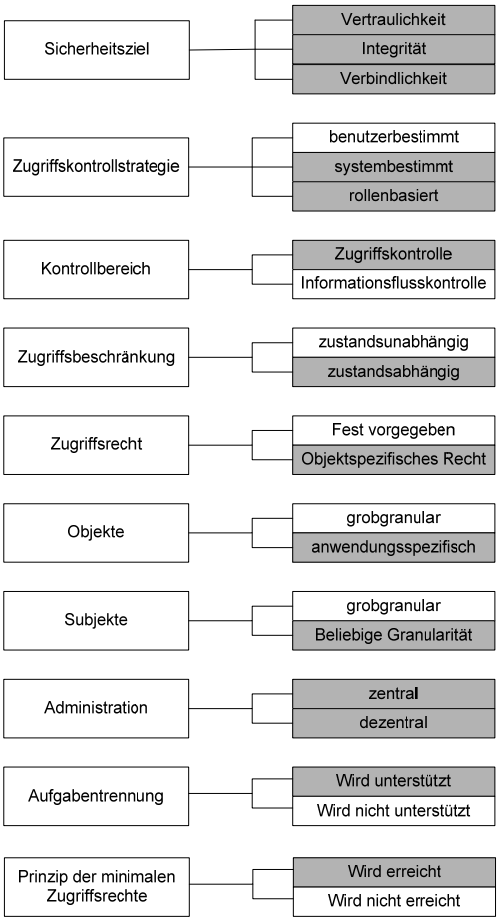

Abb. 3-4 Kriterien für die Klassifizierung und deren Ausprägungen

- Das Zugriffskontrollmodell sollte eine systemweite oder rollenbasierte Zugriffskontrollstrategie umsetzen können. Eine benutzerbestimmte Zugriffskontrollstrategie sollte nicht unterstützt werden, denn ein beliebiges Subjekt darf nicht darüber entscheiden können, welches andere Subjekt Zugriffe auf Funktionen erhält.
- Mit einem Zugriffskontrollsystem, das die Zugriffskontrolle für ein Anwendungssystem implementiert, genügt es, die direkte Zugriffskontrolle zu kontrollieren. Für eine Kontrolle des Informationsflusses besteht kein Bedarf, da durch die Geschäftslogik definiert wird, welcher Informationsfluss zwischen den einzelnen Objekten innerhalb des AwS notwendig ist.
- Bei der Zugriffsbeschränkung sollte es möglich sein, sowohl zustandsunabhängige als auch zustandsabhängige Zugriffsrechte zu verwenden.

- Das Zugriffsrecht muss ein objektspezifisches Zugriffsrecht sein. Es sollten keine durch das Zugriffskontrollmodell fest vorgegebenen Zugriffsrechte vorliegen.
- Objekte sollten anwendungsspezifisch modelliert werden können.
- Subjekte sollten möglichst feingranular modelliert werden können, damit zum einen die Verbindlichkeit sichergestellt werden kann und zum anderen es nicht zur Verletzung des Prinzips der minimalen Zugriffsrechte kommt.
- Das Zugriffskontrollmodell sollte die Möglichkeit bieten sowohl eine zentrale als auch dezentrale Administration implementieren zu können.
- Das Modell sollte über Elemente zum Modellieren der Aufgabentrennung verfügen.
- Das Prinzip der minimalen Rechte sollte durch das Modell eingehalten werden können.

3.2 Ausgewählte Zugriffskontrollmodelle

In diesem Kapitel werden ausgewählte Zugriffskontrollmodelle untersucht und ein Überblick über den Stand der Forschung gegeben. In Kapitel 3.3 werden die Zugriffskontrollmodelle in die erstellte Klassifikation eingeordnet, gegenübergestellt und bewertet. Zudem wird erläutert, welches Zugriffskontrollmodell für die Zugriffskontrolle im AwS geeignet ist.

Es wurden die folgenden Zugriffskontrollmodelle ausgewählt, die vielfach in der Praxis eingesetzt werden. Es werden die Zugriffskontrollmodelle

- Zugriffsmatrix-Modell,
- BIBA-Integritätsmodell,
- Clark-Wilson-Modell,
- Chinese-Wall Modell,
- Rollenbasiertes Zugriffskontrollmodell

und die zwei Informationsflussmodelle beschrieben:

- Bell-LaPadula-Modell,
- Verbandsmodell.

3.2.1 Zugriffsmatrix-Modell

Das Zugriffsmatrix-Modell ist das älteste Zugriffskontrollmodell und wurde als erstes Modell im Referenzmonitor für Betriebssysteme implementiert[23]. Erweiterungen und Beurteilungen finden sich u. a. in (Eckert 2012, S. 264–272; Ferraiolo et al. 2003, S. 36–38; Seufert 2001, S. 47–58).

Das Zugriffsmatrix-Modell besteht aus folgenden drei Modellkomponenten:

- Eine Menge von passiven Objekten,
- eine Menge von aktiven Subjekten,
- eine Menge an Zugriffsrechten, welche die Manipulationsmöglichkeiten der Objekte durch Subjekte ausdrücken.

Eine Zugriffsmatrix (eng. Access Control Matrix, kurz ACM) stellt die Schutzkonfiguration eines Systems dar. Objekte bilden die Spalten dieser Matrix, Subjekte die Zeilen. Die dazugehörigen Zugriffsrechte stehen im Schnittpunkt von Objekten und Subjekten dieser Matrix (Eckert 2012, S. 264–265).

Definition 3-1 Zugriffsmatrix (Eckert 2012, S. 264)

Eine $|S_t| \times |X_t|$-Matrix M_t modelliert zum Zeitpunkt t den Schutzzustand eines Systems, wobei gilt:

- Die Spalten der Matrix werden durch die Menge X_t der Objekte und
- die Zeilen der Matrix werden durch die Menge S_t der Subjekte zum Zeitpunkt t definiert.
- Es gilt: M_t: $|S_t| \times |X_t| \rightarrow 2^Z$, wobei Z die Menge der Zugriffsrechte festlegt und 2^Z die Potenzmenge der Menge Z bezeichnet. Der Eintrag $M_t(s,x) = \{z_1, ..., z_n\}$ beschreibt die Menge der Zugriffsrechte z, die das Subjekt s zum Zeitpunkt t an dem Objekt x besitzt.

Für jeden Zeitpunkt **t** modelliert die Zugriffsmatrix die gültigen Zugriffsrechte der Subjekte an Objekten eines Systems. Die Matrix $\mathbf{M_t}$, kann durch Befehle, wie Erzeugen und Löschen von Subjekten und Objekten, verändert werden. Die Matrix selbst ist ebenfalls ein zu schützendes Objekt, so dass Zugriffsrechte zur Durchführung der Änderungen modelliert werden müssen (Eckert 2012, S. 265).

[23] In der Literatur finden sich ausführliche formale Darstellungen zu diesem Modell in Lampson (1974); Graham und Denning (1971); Harrison et al. (1976); Landwehr (1981).

Der Schutzzustand einer Matrix M_t lässt sich beispielhaft anhand der Tabelle **Tab. 3-1** visualisieren.

Tab. 3-1 Beispiel einer Zugriffsmatrix

Subjekte/Objekte	Datei 1	Datei 2	Datei 3	Prozess 1
Sabine	Lesen, schreiben, Eigentümer	-	Schreiben	-
Frank	-	Ausführen	-	Sperren
Anja	-	Lesen	Lesen	-
Michael	Lesen	-	-	-

In Tab. 3-1 *hat beispielsweise Sabine das Zugriffsrecht Datei 1 zu lesen und zu schreiben, Datei 3 darf sie nur schreiben. Gleichzeitig ist vermerkt, dass Sabine der Eigentümer von Datei 1 ist. Frank darf die Datei 2 ausführen und den Prozess 1 sperren, während Anja nur lesenden Zugriff auf die Dateien 2 und 3 besitzt. Michael besitzt nur Lesezugriff auf Datei 1.*

3.2.1.1 Vergabe der Zugriffsrechte

Die Zugriffskontrollstrategie des Zugriffsmatrix-Modells ist benutzerbestimmt. Die Erlaubnis der Rechtevergabe und des Rechteentzugs kann je nach Implementierung auf bestimmte Subjekte beschränkt werden. Es sind drei Prinzipien unterscheidbar:

- **Eigentümerprinzip**: Das Subjekt **s** muss zum Zeitpunkt **t** der Eigentümer des Objektes **o** sein und damit das Zugriffsrecht „Eigentümer“ besitzen. Das Eigentümerprinzip ist vor allem in Dateisystemen zu finden. Die Implementierung sieht vor, dass der Erzeuger einer Datei automatisch der Eigentümer einer Datei wird. Das Eigentümer-Recht kann anderen Subjekten aus der Menge **S** übertragen werden.
- **Besitzerprinzip**: Voraussetzung für die Weitergabe eines Zugriffsrechts ist bereits der Besitz des Zugriffsrechts. Das Besitzerprinzip ist weniger restriktiv.
- **Administratorenprinzi**p: Die Weitergabe und der Entzug von Zugriffsrechten werden auf eine Gruppe von Administratoren beschränkt (Seufert 2001, S. 50).

Die Vergabe bzw. Weitergabe von Zugriffsrechten ist ein Schwachpunkt der Zugriffsmatrix, vor allem dann, wenn das Besitzerprinzip zum Einsatz kommt, da jeder, der ein bestimmtes Zugriffsrecht besitzt, dies auch weitergeben kann.

3.2.1.2 Verwaltung und Implementierung der Zugriffsmatrix

Aus theoretischer Sicht ist die Zugriffsmatrix ein sehr interessantes Konstrukt, aber bei einer großen Anzahl von Subjekten und Objekten wird die Matrix sehr groß und in der Regel sind nur wenige Zellen besetzt. Es kann zwischen statischen Implementierungen der Zugriffsmatrix, z. B. Paketfilter bei Firewalls und dynamischen, z. B. in Betriebssystemen, unterschieden werden. Eine dynamische Zugriffsmatrix wird auch für die theoretische Diskussion von Sicherheitsproblemen herangezogen. Die Zustandsübergänge, also die Veränderungen der Matrix, werden durch Definition von Grundbefehlen modelliert. Mittels dieser Befehle wird untersucht, ob ein Zugriffskontrollsystem, das auf der Zugriffsmatrix basiert, sicher arbeitet (Graham und Denning 1971).

Da zu Beginn des praktischen Einsatzes die Verwaltung einer dünn besetzten Zugriffsmatrix noch ineffizient war, gab es kaum Implementierungen der vollständigen Matrix. Um die große Anzahl von leeren Zellen zu minimieren, wird nicht die Matrix abgespeichert, sondern linear verkettete Listen. In Computersystemen sind deshalb Fähigkeitslisten (capability lists) oder Zugriffskontrolllisten (access control lists, kurz ACL) implementiert (Anderson 2001, S. 54–55; Saltzer und Schroeder 1975).

Fähigkeitslisten stellen die Zugriffsrechte eines Subjektes dar und sind die Zeilen der Zugriffsmatrix. Jedes Subjekt wird mit einer Fähigkeitsliste (siehe **Tab. 3-2**) verbunden, in der alle erlaubten Zugriffe auf die entsprechenden Objekte gespeichert sind. Die Fähigkeitslisten werden im Stammdatensatz der Nutzerverwaltung hinterlegt. Sie eigenen sich auch für verteilte Systeme durch die Möglichkeit der Weitergabe der Fähigkeitslisten, wobei zu beachten ist, das diese vor Manipulationen zu schützen sind (Seufert 2001, S. 51).

Tab. 3-2 Eine Fähigkeitsliste, abgeleitet aus der Zugriffsmatrix in **Tab. 3-1**

Subjekt		
Sabine	Datei 1: lesen, schreiben, Eigentümer	Datei 3: Schreiben
Frank	Datei 2: Ausführen	Prozess 1: Sperren
Anja	Datei 3: Lesen	Datei 3: Lesen
Michael	Datei 1: Lesen	

Mit der Fähigkeitsliste kann schnell über das Zugriffsrecht eines Subjektes entschieden werden. Der Nachteil ist jedoch, dass sich die Frage, welche Subjekte auf ein bestimmtes Objekt Zugriff haben, nur durch sequentielles Durchsuchen aller Fähigkeitslisten beantworten lässt (Seufert 2001, S. 51–52).

Zugriffskontrolllisten hingegen stellen die Spalten der Zugriffsmatrix dar und sind Listen, in denen pro Objekt hinterlegt wird, welche Subjekte welche Zugriffsrechte besitzen. Die Zugriffsliste wird mit dem entsprechenden Objekt verbunden und z. B. mit den Eigenschaften einer Datei gespeichert. Zugriffskontrolllisten werden u. a. innerhalb des Betriebssystems Windows 2000 (Kuppinger 2000, S. 411) verwendet.

Tab. 3-3 Eine Zugriffskontrollliste, abgeleitet aus der Zugriffsmatrix in **Tab. 3-1**

Objekt		
Datei 1	Sabine: Lesen, schreiben, Eigentümer	Michael: Lesen
Datei 2	Frank: Ausführen	Anja: Lesen
Datei 3	Sabine: Schreiben	Anja: Lesen
Prozess 1	Frank Sperren	

Der Vorteil, schnell über ein Zugriffsrecht entscheiden zu können, bringt den Nachteil, dass die Menge der Objekte, für die einem Subjekt der Zugriff gewährt wird, nicht effizient ermittelt werden kann. Da der primäre Nutzen der Zugriffskontrolllisten, die schnelle Entscheidung über das Zugriffsrecht ist, wird dieser Nachteil akzeptiert (Eckert 2012, S. 638).

3.2.1.3 Sicherheitsfrage

In der Forschung wurde die Sicherheitsfrage (Harrison et al. 1976) mit Hilfe von vorgegebenen Befehlen **B** des HRU-Schemas[24] überprüft. Die untersuchte Fragestellung, das sog. Safety-Problem, lautet, ob ausgehend von einem gegebenen Schutzzustand M_t ein Subjekt **s** das Zugriffsrecht **z** an dem Objekt **x** erhalten kann, wenn es dieses Zugriffsrecht im Zustand M_t noch nicht besitzt. Es muss gezeigt werden, dass ausgehend von Zustand M_t, mit $z \notin M_t(s,x)$, ein Zustand M_t' erreicht werden kann, für den gilt $z \in M_t'(s,x)$.

Die Ergebnisse lassen sich wie folgt zusammenfassen:

1. Falls jeder Befehl $b \in B$ eines HRU-Schemas nur Elementaroperationen beinhaltet, d.h. mono-operational ist, dann ist die Sicherheitsfrage für ein allgemeines Zugriffsrecht $z \in Z$ entscheidbar und es existiert ein Algorithmus.
2. Die Unentscheidbarkeit der Sicherheitsfrage konnte bewiesen werden, indem die Sicherheitsfrage auf das unentscheidbare Halteproblem von Turing-Maschinen (Hopcroft et al. 2002, S. 350) zurückführt wurde. „Die Unentscheidbarkeitsaussage besagt, dass es keinen Algorithmus gibt, der bei einer gegebenen Anfangs-

[24] HRU bildet sich aus den Anfangsbuchstaben der Namen Harrison, Ruzzo und Ullmann.

konfiguration eines Systems mit einer beliebigen festgesetzten Menge von Kommandos in endlicher Zeit entscheiden kann, ob ein Recht **z** in den Besitz des Subjektes **s** gelangen kann“ (Eckert 2012, S. 269). Es wurde damit bewiesen, dass die Sicherheitsfrage NP-vollständig[25] ist (Harrison et al. 1976).

Die Sicherheitsfrage ist für beliebige Systeme nicht entscheidbar, d. h. es existiert kein genereller Algorithmus. Dies heißt jedoch nicht, dass für ein konkretes System mit konkret festgelegter Kommando-Menge diese Frage nicht beantwortet werden kann. Für konkrete Systeme kann ein entsprechendes Verfahren konstruiert werden. Solche Modelle sind z. B. Take-Grant-Modelle[26]. Diese untersuchen eine Zugriffsmatrix mit bestimmten Nebenbedingungen. Die Untersuchungen zeigten, dass ein Entscheidungsverfahren existiert, mit dem die Entscheidbarkeit der Sicherheitsfrage geklärt werden kann (Eckert 2012, S. 269; Seufert 2001, S. 66–72).

3.2.1.4 Gruppenkonzept

Eine zusätzliche Möglichkeit, die Implementierungen als Fähigkeitslisten bzw. Zugriffskontrolllisten zu verkleinern, ist das Einführen von Subjektgruppen zur Bündelung von Zugriffsrechten. Der Vorteil von Gruppen ist, dass sich der Umfang der Matrix dadurch verkleinert. Nachteilig ist jedoch, dass es nicht möglich ist, gezielt für einzelne Subjekte, unabhängig von ihrer Gruppe Zugriffsrechte zu geben. Werden anderen Subjekten der eigenen Gruppe gewisse Zugriffsrechte gewährt, so gelten diese Zugriffsrechte für alle Subjekte derselben Gruppe. „Diese Einschränkung ist ein wesentlicher Schwachpunkt einer solchen Gruppeneinteilung; sie kann einer sinnvollen Vergabe von Zugriffsrechten nicht zu vernachlässigende Schwierigkeiten bereiten“ (Weck 1993, S. 175).

3.2.1.5 Einordnung und Diskussion

Zunächst erfolgt die Einordnung in die Klassifikation (siehe Kapitel 3.1.3):

- Das verfolgte Sicherheitsziel des Zugriffsmatrixmodells ist die Integrität.
- Die Zugriffskontrollstrategie ist benutzerbestimmt.
- Der Kontrollbereich überprüft die Zugriffskontrolle.

[25] Eine ausführliche Erklärung und Beschreibung findet sich in Hopcroft et al. (2002), Kapitel 10.

[26] Durch die Nebenbedingungen und Einschränkungen der Zugriffsmatrix erreichten die Take-Grant-Modelle keine praktische Bedeutung und es wird auf die Literatur verwiesen Bishop (1981); Bishop und Snyder (1979); Lipton und Snyder (1977); Snyder (1981).

- Die Zugriffsrechtebeschränkung ist zustandsunabhängig; allein das Zugriffsrecht reicht aus, um Zugriff zu erhalten. Es existieren jedoch Erweiterungen, die zustandsabhängige Zugriffsrechtebeschränkung vorsehen.
- Die Zugriffsrechte werden universell festgelegt.
- Die Granularität der Objekte ist von der jeweiligen Anwendung abhängig.
- Die Subjekte können je nach Anwendungsfall als Gruppen oder als einzelne Subjekte festgelegt werden.
- Vom Modell her ist die Administration dezentral angelegt. Die Administration kann zentral den Rahmen vorgeben, aber innerhalb dieses Rahmens kann jedes Subjekt die Zugriffsrechte dezentral weitergeben.
- Die Modellierung der Aufgabentrennung wird vom Grundmodell nicht unterstützt.
- Das Prinzip des minimalen Rechts kann durch das Grundmodell nicht immer eingehalten werden. Es ist zum einen wegen der universellen Zugriffsrechte und zum anderen durch die entstehende Komplexität in Bezug auf die Granularität der Objekte und Subjekte schwer überprüfbar. Durch das aus Vereinfachungsgründen fast immer angewandte Gruppenkonzept besteht ebenfalls die Möglichkeit, dieses Prinzip zu verletzen.

Das Zugriffsmatrix-Modell bzw. dessen Implementierungen als Referenzmonitor finden immer noch Verwendung. Es ist ein sehr gut untersuchtes Zugriffskontrollmodell. Aber eine Schwachstelle ist, dass die Zugriffsmatrix anfällig für Trojanische Pferde ist, weil diese die Zugriffsrechte ändern können, ohne dass ein Subjekt dies explizit bemerkt (Sandhu 1992, S. 123). Eine streng typisierte Zugriffsmatrix, bei der sowohl Objekte als auch Subjekte von vorgegebenen Typen abgeleitet werden, ist eine Möglichkeit dies zu verhindern.[27]

Für das Grundmodell der Zugriffsmatrix wurden viele Erweiterungen diskutiert. Um beispielsweise einen flexibleren Einsatz zu gewährleisten, werden Subjekte und Objekte mit Attributen versehen, die zur Laufzeit ausgewertet werden (Zhang et al. 2005). Andere Überlegungen beziehen sich auf die Verbesserungen der Bedienbarkeit durch den Nutzer (Cao und Iverson 2006).

[27] Für die ausführliche Besprechung diese Problems wird auf die Literatur verwiesen Sandhu (1992).

3.2.2 Bell-LaPadula-Modell

Das Bell-LaPadula Modell wurde von 1973 bis 1976 im Auftrag des amerikanischen Militärs entwickelt und gilt als eines der ersten vollständig formalisierten Zugriffskontrollmodelle (Bell und LaPadula 1973b; Bell und LaPadula 1973a). "The Report was to describe a mathematical model of security in computer systems" (Bell 2005). Mit Hilfe eines formalen Modells wurde bewiesen, dass das Bell-LaPadula Modell zu sicheren Zugriffskontrollsystemen führen kann. Das Modell regelt den Informationsfluss zwischen Objekten und Subjekten aufgrund von Zugriffsklassen.

3.2.2.1 Grundlagen des Bell-LaPadula Modells

Die Basis des Bell-LaPadula-Modells bildet eine Zugriffsmatrix. Die Menge der festgelegten Zugriffsrechte umfasst: **{read, write, append, execute, control}**. **{read, write, append, execute}** beziehen sich auf Operatoren für Objekte in Betriebssystemen. **Control** bezieht sich auf die Zugriffsmatrix selbst und berechtigt zur Weitergabe bzw. zum Entzug von Zugriffsrechten (Bell und LaPadula 1973b, S. 13, 24).

Zugriffsklassen, Sicherheitskategorien und Sicherheitsklassen

Neben der Zugriffsmatrix wird im Bell-LaPadula Modell eine geordnete Menge von Zugriffsklassen (ZK) als Sicherheitslevels eingeführt. Ein Sicherheitslevel ist ein einer Entität zugeordnetes hierarchisches Attribut, das den Grad der Empfindlichkeit angibt (Amoroso 1994, S. 68). Zugriffsklassen dienen dazu, Subjekte und Objekte in ihrer Vertraulichkeitsstufe zu klassifizieren und damit die Vertraulichkeit der Dokumente sicherzustellen. In der Literatur wird oftmals die Zugriffsklasse der Objekte **classification**, die der Subjekte **clearance** genannt. In dieser Arbeit wird der allgemeine Begriff Zugriffsklasse sowohl für Objekte als auch für Subjekte verwendet (Gasser 1988, S. 64). Für diese Zugriffsklassen wird folgende Ordnung festgelegt:

$$\textbf{Unklassifiziert} \leq \textbf{vertraulich} \leq \textbf{geheim} \leq \textbf{streng geheim}.$$

Neben Zugriffsklassen existieren im Modell Sicherheitskategorien, die eine nicht hierarchische Menge von Objekten und Subjekten darstellen:

{Militär, Nato, Nuklear, Nasa}

Zugriffsklassen und Sicherheitskategorien werden zu Sicherheitsklassen (SK) zusammengefasst, die wie folgt definiert sind:

SK = {(geheim, Ø), (vertraulich, Ø), ...(vertraulich, {Militär}) , (vertraulich, {Nato}) ... (vertraulich, {Nato, Nuklear})}

Für diese Sicherheitsklassen wird eine Ordnungsrelation festgelegt:

(geheim, Ø) $\geq$ (vertraulich, Ø)

(vertraulich, {Nato, Nuklear}) $\geq$ (vertraulich, {Nato})

Für die Überprüfung des Zugriffs wird eine Dominanzrelation[28] über die Sicherheitsklassen definiert, die transitiv, reflexiv und antisymmetrisch ist.

SK_1 = (ZK_1, Menge von $Sicherheitskategorie_1$)
dominiert
SK_2 = (ZK_2, Menge von $Sicherheitskategorie_2$)

dann und nur dann, wenn

ZK1 $\geq$ ZK2 ist
und
Menge der Sicherheitskategorie1 $\supseteq$ die Menge der Sicherheitskategorie2.

Zugriffsklassen und Sicherheitsklassen gelten systemweit. Um zu verhindern, dass Unberechtigte in Kenntnis von Informationen gelangen, wurden die folgenden zwei Regeln eingeführt (Bell und LaPadula 1973b; Bell und LaPadula 1973a): **Simple Security Regel** (no-read-up Regel) und ***-Eigenschaft** (no-write-down Regel).

Abb. 3-5 zeigt die grafische Darstellung der Simple Security Regel und **Abb. 3-6** stellt die *-Eigenschaft jeweils in einem Level-Diagramm dar. In einem Leveldiagramm werden die horizontalen Linien als Grenzen zwischen den verschiedenen Sicherheitsstufen interpretiert (Amoroso 1994, S. 102–104).

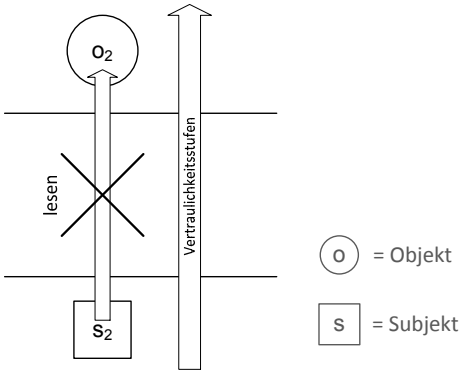

Abb. 3-5 Simple Security Regel nach Bell-La-Padula nach (Seufert 2001, S. 87)

[28] Eine Definition der Dominanzrelation findet sich in Amoroso (1994, S. 74–76).

Die **Simple Security Regel** bestimmt, dass einem Subjekt **s** nur dann ein Leserecht auf einem Objekt **o** gewährt ist, wenn **s** das entsprechende Zugriffsrecht in der Zugriffsmatrix $\mathbf{M_t\,(s,o)}$ besitzt und die Zugriffsklasse des Subjektes größer oder gleich der Zugriffsklasse des Objektes ist. Es gilt: $\mathbf{z \in M\,(s,o)\ und\ sc\,(s) \geq sc\,(o)}$ (Bell und LaPadula 1973a, S. 6–8).

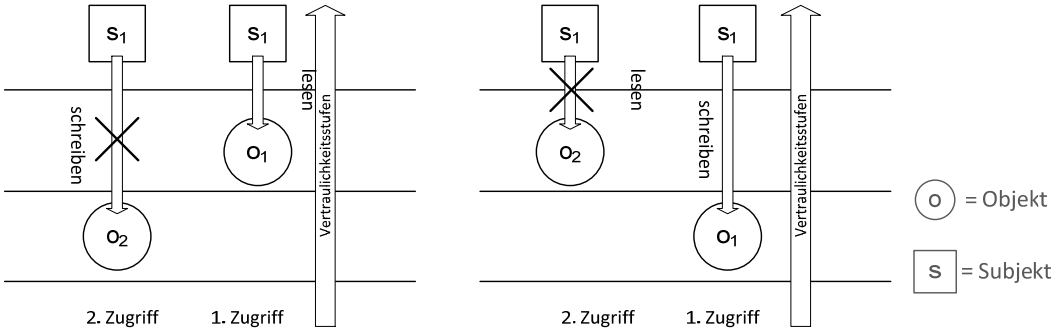

Abb. 3-6 *-Eigenschaft nach Bell La-Padula nach (Amoroso 1994, S. 105; Seufert 2001, S. 87)

Die ***-Eigenschaft** legt fest, dass ein schreibender Zugriff auf ein Objekt **o** durch ein Subjekt **s** nur zulässig ist, wenn die Zugriffsklasse des Objektes mindestens so hoch ist wie die Zugriffsklasse des Subjektes. Es gilt: $\mathbf{schreiben \in M\,(s,o) \wedge sk(s) \leq sk(o).}$ Ein Lese-Schreib-Zugriff auf ein Objekt **o** ist nur möglich, wenn die Zugriffsklasse des Subjektes gleich der Zugriffsklasse des Objektes ist, d.h. es gilt: $\mathbf{read\text{-}write \in M\,(s,o) \wedge sk(s) = sk(o).}$

Zu beachten ist, dass sich diese Regel immer auf zwei aufeinander folgende Zugriffe bezieht. Wird im ersten Zugriff ein Objekt mit der Klassifizierung geheim zum Lesen geöffnet, so wird der in einem späteren Schritt angeforderte Schreib-Zugriff auf ein Objekt mit der Zugriffsklasse unklassifiziert verweigert, da es beim zweiten Zugriff möglich ist, geheime Inhalte in ein unklassifiziertes Objekt zu schreiben. Umgekehrt muss nach dem ersten schreibenden Zugriff auf ein unklassifiziertes Objekt der spätere Zugriff auf ein höher klassifiziertes Objekt zum Lesen unterbunden werden (Bell und LaPadula 1973a, S. 6–8).

Um die Informationssicherheit zu gewährleisten, müssen beide Regeln implementiert werden. Ein Nachteil bei der Umsetzung der *-Eigenschaft ist, dass sich viele Vergleichsoperationen im Betriebssystem ergeben, da für jedes neu angeforderte Objekt die Sicherheitsklassen dieses Objektes mit jedem Objekt, auf das das Subjekt im Augenblick Zugriff hat, verglichen werden müssen. Zur Vereinfachung wird bei der

Implementierung die gegenwärtige Sicherheitsklasse in einer Systemvariablen gespeichert. Vor dem Zugriff auf ein neues Objekt ist damit nur ein Vergleich erforderlich (Bell und LaPadula 1973a, S. 12–21; Amoroso 1994, S. 104–106).

Zusätzlich wurde zur leichteren Implementierung des Modells das Konzept der vertrauenswürdigen Subjekte eingeführt. Ein vertrauenswürdiges Subjekt garantiert keinen sicherheitsverletzenden Informationstransfer zu vollziehen. Bei vertrauenswürdig eingestuften Subjekten wird die *-Eigenschaft nicht überprüft. Vertrauenswürdige Subjekte sind z. B. Betriebssystemprozesse, die gut getestet sind und von denen bekannt ist, dass keiner ihrer Subprozesse die *-Eigenschaft verletzt (Amoroso 1994, S. 118–119; Bell und LaPadula 1976, S. 64–76).

3.2.2.2 Einordnung und Diskussion

Zunächst erfolgt die Einordnung des Bell-LaPadula-Modells in die Klassifikation (siehe Kapitel 3.1.3), anschließend werden ausgewählte Schwächen aufgezeigt:

- Das Bell-LaPadula-Modell setzt das Sicherheitsziel Vertraulichkeit um (Gasser 1988, S. 68).
- Die Zugriffsklassen sind für die gesamte Organisation definiert und dadurch wird die benutzerbestimmte der Zugriffsmatrix zu einer systemweiten Zugriffskontrollstrategie.
- Der Kontrollbereich des Modells ist der Informationsfluss.
- Die Zugriffsbeschränkung hängt nur vom Modell ab und ist damit zustandsunabhängig.
- Die Zugriffsrechte sind fest vorgegeben und in zwei Klassen eingeteilt - die lesenden und schreibenden Zugriffe.
- Die Granularität der Objekte und Subjekte sind vom Modell nicht vorgeschrieben und damit beliebig. Da die Basis eine Zugriffsmatrix ist, entsteht ebenfalls wieder das Problem bei der Bildung von Nutzergruppen.
- Die Administration wird zentral durchgeführt, da eine systemweite Zugriffskontrollstrategie umgesetzt werden soll.
- Die Umsetzung der Aufgabentrennung wird vom Modell nicht unterstützt.
- Das Modell macht keine Aussage zur Berücksichtigung des Prinzips der minimalen Zugriffsrechte.

Das Bell-LaPadula-Modell sagt nichts darüber aus, wie mit einer Änderung der Sicherheitsklasse des Subjekts oder des Objektes umgegangen werden soll. Auch das 4-Augenprinzip kann mit diesem Modell nicht abgebildet werden (McLean 1990, S. 2). Neben diesen beiden Schwächen hat das Bell-LaPadula-Modell u. a. noch zwei weitere, die durch die beiden Regeln verursacht werden: Blindes Schreiben und Entferntes Lesen.

Blindes Schreiben

Im Bell-LaPadula Modell gibt es keine Regel, die das Schreiben von einer niedrigeren Sicherheitsklasse in eine höhere verbietet. Dadurch kann ein niedriger klassifiziertes Subjekt in ein höher klassifiziertes Objekt schreiben. Da diesem Subjekt jedoch verboten ist, ein höher klassifiziertes Objekt zu lesen, kann es eine Änderung nicht verifizieren. Durch die fehlende Kontrolle des Schreibvorgangs besteht die Gefahr der Integritätsverletzung. Sollte die Veränderung unbeabsichtigt oder bösartig vorgenommen worden sein, kann das schreibende Subjekt dies nicht bemerken. Eine Lösung wäre die no-write-down Regel zu verschärfen, dass nur auf derselben Sicherheitsstufe geschrieben werden darf (Amoroso 1994, S. 114–116).

Entferntes Lesen

In einem verteilten System, in dem die Bell-La-Padula Regeln angewendet werden, kann es zu Schwierigkeiten kommen, wenn ein Lesezugriff auf einem entfernten System stattfinden soll. Ein Rechner **A** und ein Subjekt sind als geheim eingestuft, während der Rechner **B** als vertraulich eingestuft ist. Das Subjekt auf dem Rechner **A** greift nun lesend auf den Rechner **B** zu. Dieser Informationsfluss ist nach Bell-La-Padula zulässig, da vertraulich < geheim. Um den entfernten lesenden Zugriff zu ermöglichen, muss eine Verbindung zwischen Rechner **A** und **B** aufgebaut werden. Dazu sendet **A** eine Anforderung zur Etablierung einer Verbindung an den Rechner **B**. Problematisch ist nun, dass diese Anfrage von Rechner **A** eine Schreib-Operation auf dem Rechner **B** zur Folge hat. Es liegt eine write-down Operation von **A** nach **B** vor, dies verletzt die *-Eigenschaft des Bell-La-Padula Modells. Zur Lösung des Problems sind solche Anfragenachrichten zuzulassen, aber speziell zu kontrollieren, um sicherzustellen, dass kein unzulässiger Informationsfluss auftreten kann (Amoroso 1994, S. 117–118).

3.2.3 BIBA-Integritätsmodell

Das Biba-Integritätsmodell (kurz Biba-Modell) nimmt als Basis das Bell-LaPadula-Modell und beschreibt systemweite Regeln, um die Integrität zu sichern und damit die Integritätsprobleme des Bell-LaPadula-Modells zu beheben (Amoroso 1994, S. 134–145). Die Regeln des Bell-LaPadula-Modells werden im Biba-Modell umgekehrt.

3.2.3.1 Beschreibung des Modells

Beim Biba-Modell werden die Objekte und Subjekte in Zugriffsklassen eingeteilt und die Sicherheitslevels werden gebildet, um das Sachziel Integrität zu gewährleisten. Die Höhe der Integritätsebene der Subjekte und Objekte wird bestimmt in Abhängigkeit davon, wie schwerwiegend eine Verletzung der Integrität ist. Subjekte bzw. Objekte werden in eine niedrigere Integritätsebene eingestuft, wenn eine geringere Wichtigkeit der Integrität vorhanden ist. Auch in diesem Modell wird eine Dominanzrelation zum Vergleich zweier Sicherheitsklassen verwendet (Amoroso 1994, S. 136).

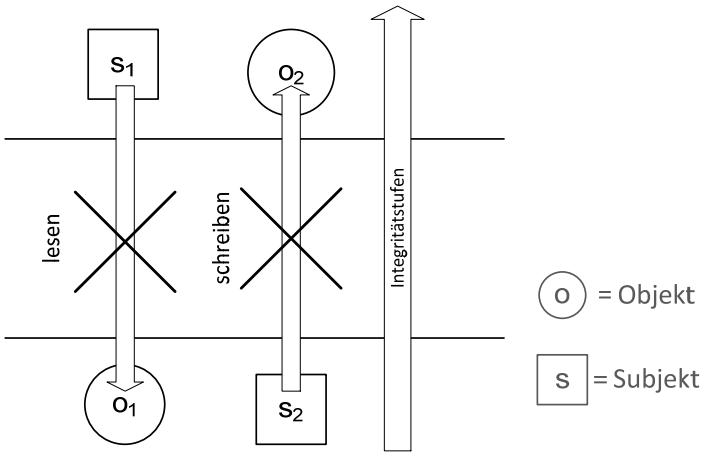

Abb. 3-7 Biba-Regeln: no-read-down und no-write-up nach (Amoroso 1994, S. 137)

Die folgenden zwei Regeln des Biba-Modells sind in **Abb. 3-7** grafisch dargestellt:

- **No-read-down**: Subjekte können keine Informationen in Objekten lesen, die einen geringeren Integritätslevel besitzen. Damit soll verhindert werden, dass die Information eines Objektes, das sich auf einem niedrigeren Integritätslevel befindet, die Integrität des Subjektes beschädigt.
- **No-Write-up**: Mit dieser Regel ist es Subjekten verboten, in Objekte mit einem höheren Integritätslevel zu schreiben.

Varianten der Biba-Regeln

Es existieren u. a. Varianten der Biba-Regeln, die im Folgenden kurz beschrieben werden und eine Aufweichung der no-read-down- bzw. no-write-up-Regel darstellen:

- **Subjekt-read-down**: In diesem Modell ist es Subjekten erlaubt, Informationen aus Objekten mit einer niedrigeren Integritätsstufe zu lesen. Dieses Lesen hat zur Folge, dass die Integritätsstufe des Subjektes jener des Objektes angepasst wird. Es wird das Subjekt deklassiert.
- **Subject-write-up**: Hier ist es Subjekten erlaubt, in Objekte mit höherer Integrität zu schreiben. Dies hat dann eine Abstufung der Sicherheitsklasse des Objektes zur Folge (Amoroso 1994, S. 137–138).

Bei einer Aufweichung der Biba-Regeln ist zu bedenken, dass beim Lese- bzw. Schreib-Zugriff alles auf einer Ebene stattfindet, es findet eine Deklassierung des Subjektes bzw. Objektes statt. Regeln zum Erhöhen der Integritätsstufen existieren nicht. Damit ist sehr genau abzuwägen, ob diese Varianten des Modells in der Praxis angewandt werden (Amoroso 1994, S. 139).

Ein Beispiel für den Einsatz des Biba-Modells sind medizinische Geräte zur Erstellung eines Elektrokardiogramms. Dieses hat zwei Modi: Kalibrierung und Nutzung. Die Kalibrierungsdaten müssen vor Veränderungen durch normale Nutzer geschützt werden. Diese Nutzer müssen zwar die Daten lesen können, dürfen aber die Kalibrierungsdaten nicht verändern. Wenn der Nutzer das Gerät zurücksetzt, gehen die aktuellen Patientendaten zwar verloren, die Kalibrierungsdaten bleiben jedoch unverändert (Anderson 2001, S. 145).

3.2.3.2 Einordnung und Diskussion

Die Einordnung in die Klassifikation ergibt folgendes Ergebnis:

- Das Sicherheitsziel des Biba-Modells ist die Integrität.
- Die Zugriffskontrollstrategie ist eine systemweite, da es die benutzerbestimmte Strategie der Zugriffsmatrix um eine systemweite Komponente erweitert.
- Das Biba-Modell überprüft die Zugriffskontrolle.
- Die Zugriffsbeschränkung ist zustandsunabhängig, da sie nur vom Modell abhängt.

- Die Zugriffsrechte sind wie beim Bell-LaPadula-Modell fest in schreibende und lesende Zugriffsrechte eingeteilt.
- Die Objekte und Subjekte sind von beliebiger Granularität.
- Die Administration wird zentral durchgeführt.
- Die Aufgabentrennung und das Prinzip der minimalen Zugriffsrechte werden vom Modell nicht berücksichtigt.

Die im Biba-Modell definierten Regeln sind „sehr restriktiv und wenig flexibel, so dass das Modell nur stark eingeschränkt einsetzbar ist“ (Eckert 2012, S. 291). Außerdem stellt sich die Frage, ob sich Integrität in sinnvolle Sicherheitslevels einordnen lässt, da eine Integritätsverletzung immer schwerwiegend ist.

3.2.4 Verbandsmodell

Das Verbandsmodell (engl. lattice model) befasst sich mit der Kontrolle des Informationsflusses zwischen Datenvariablen einer Anwendungssoftware (Denning 1976). Hierbei wird nicht der Zugriff auf Objekte beschränkt, sondern der Umgang mit der Information, die durch Objekte repräsentiert werden, wird reglementiert, indem zulässige und unzulässige Informationskanäle festgelegt werden (Eckert 2012, S. 292–293). Die Objekte können sehr feingranular sein, bis hin zu Programmvariablen. Die Regelungen des Verbandsansatzes werden direkt in den Programmcode eines zu sichernden Anwendungssystems eingebunden. Die Sicherheitsüberprüfungen können während der Laufzeit oder bei der Übersetzung des Programms erfolgen. Ändern sich die Sicherheitsklassen der Objekte zur Laufzeit nicht, kann die gesamte Überprüfung zur Übersetzungszeit berechnet werden. Durch eine Kontrolle während der Übersetzung des Programms können aufwendige Laufzeitkontrollen vermieden werden (Denning 1976, S. 239–240; Seufert 2001, S. 83).

3.2.4.1 Grundlagen und formale Beschreibung

Das Modell basiert auf der mathematischen algebraischen Struktur des Verbandes.[29] Eine Verbandsstruktur regelt die Informationsweitergabe zwischen Objekten, die als Datenbehälter angesehen werden. Die Struktur des Verbandes wird aus der Dominanzrelation zwischen Sicherheitsklassen (siehe Kapitel 3.2.2.1) gewonnen. Zum Visualisieren der Dominanzrelationen eignet sich das sog. Hasse-Diagramm:

[29] Die mathematischen Grundlagen eines Verbandes finden sich u.a. in Scheinerman (2000, S. 442–444).

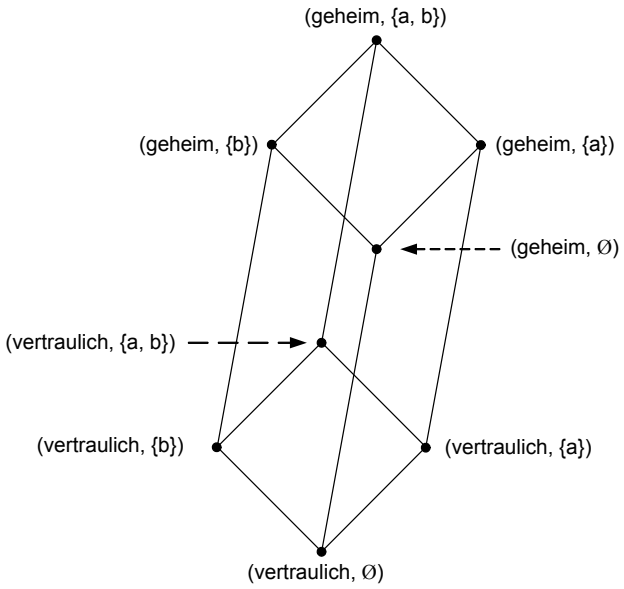

Abb. 3-8 Hasse-Diagramm (Eckert 2012, S. 294)

Gegeben seien folgende Sicherheitsklassen mit der partiellen Ordnung:

> **SK = {(geheim, Ø), (geheim, {a}), (geheim, {b}), (geheim, {a, b}), (vertraulich, Ø), (vertraulich, {a}), (vertraulich, {b}), (vertraulich, {a, b})**.

Das zugehörige Hasse-Diagramm ist in Abb. **3-8** visualisiert. Hierbei wird unter anderem

> als **Supremum: (geheim {a})** $\oplus$ **(geheim {b}) = (geheim {a, b})** und
>
> als **Infimum: (geheim {a})** $\otimes$ **(geheim {b}) = (geheim, Ø)** festgelegt**.**

Die Sicherheitsklassen werden durch ausgefüllte Kreise, das Vorliegen einer Dominanzrelation durch eine ungerichtete Kante dargestellt. Eine Kante zwischen zwei Sicherheitsklassen wird von der größeren Sicherheitsklasse am oberen Ende zur Kante der kleineren am unteren Ende gezogen. Das heißt die obere Sicherheitsklasse dominiert die untere Sicherheitsklasse. Die Informationen können von einem Objekt, das mit der Sicherheitsklasse A markiert ist, zu einem Objekt fließen, das mit der Sicherheitsklasse B markiert ist, wenn gilt $A \leq B$. Niedrig eingestufte Informationen können damit uneingeschränkt fließen (Eckert 2012, S. 293). Ein Schema im Sinne des Verbandsmodells ist genau dann sicher, wenn es keine Folge von Operationen auf Objekten $\mathbf{o} \in \mathbf{O}$ gibt, die einen Informationsfluss hervorrufen, der die Dominanzrelation verletzt, also kein abwärts gerichteter Informationsfluss existiert.

3.2.4.2 Durchsetzung der Sicherheitsbedingung

Durch die Transitivität der Dominanzrelation ≤ kann ein jeder impliziter Fluss von einer Variablen **X** zu einer Variablen **Y**, im Zeichen $X \dashrightarrow Y$, das aus einer Folge von Flüssen

$$X = Z_0 \dashrightarrow Z_1 \rightarrow \ldots \rightarrow Z_n = Y$$

resultiert, zulässig sein, wenn jeder direkte Fluss von $Z_i \dashrightarrow Z_{i+1}$ zulässig ist. Das bedeutet, dass es ausreicht, die direkten Flüsse zu kontrollieren, um die Zuverlässigkeit von Informationsflüssen zu überprüfen. Übertragen auf Programme bedeutet dies, dass eine Folge von Anweisungen zulässig ist, falls jede einzelne Anweisung zulässig ist. Die Eigenschaft des Verbandes der Existenz eines Supremums und Infimums für je zwei Verbandselemente kann ebenfalls benutzt werden, um den Kontrollaufwand zu reduzieren (Eckert 2012, S. 294).

3.2.4.3 Einordnung und Diskussion

Die Einordnung erfolgt anhand der Klassifikation aus Kapitel 3.1.3:

- Das unterstützte Sachziel der Informationssicherheit ist die Vertraulichkeit.
- Die Zugriffskontrollstrategie ist systemweit.
- Der vom Verbandsmodell unterstützte Kontrollbereich ist die Informationsflusskontrolle.
- Die Zugriffsbeschränkung ist alleine vom Modell abhängig und damit zustandsunabhängig.
- Welche Objekte, Subjekte und Zugriffsrechte betrachtet werden, wird vom Modell nicht vorgegeben. Es ist alleine abhängig von den zu kontrollierenden Informationsflüssen und kann beliebig granular gestaltet werden.
- Es wird eine zentrale Administration gefordert.
- Es wird vom Modell weder die Aufgabentrennung noch das Prinzip der minimalen Rechte unterstützt.

Für kommerzielle Aufgabenfelder findet sich kaum ein Anwendungsgebiet. Zudem greift es in die Anwendungsentwicklung ein und vermischt Geschäftslogik und Zugriffskontrolle.

3.2.5 Clark-Wilson Modell

Das Clark-Wilson-Modell wurde für den kommerziellen Bereich entwickelt. Es geht von zwei Anforderungen aus, die für die Integrität der Daten in der Betriebswirtschaft sorgen:

- die wohlgeformte Transaktion (engl. well-formed transaction) und
- die Aufgabentrennung (siehe Kapitel 2.4.3).

Diese beiden Anforderungen wurden mit dem Clark-Wilson Modell in ein semiformales Zugriffskontrollmodell für kommerzielle Anwendungssysteme überführt (Clark und Wilson 1987, S. 186–187).

Beispiel für eine wohlgeformte Transaktion: In der Buchhaltung muss bei der Buchung eines Geschäftsvorganges immer eine Soll- und Habenbuchung erfolgen. Diese Buchungen müssen innerhalb einer Transaktion ablaufen.

Beispiel für Aufgabentrennung: Wird eine Überweisung von mehr als 100 000 Euro von einem Bankangestellten angewiesen, dann erfolgt die Buchung erst, wenn ein zweiter Bankangestellter dies geprüft und gegengezeichnet hat.

Das Clark-Wilson Modell will eine systemweite Zugriffskontrolle sicherstellen und versucht, die oben genannten betriebswirtschaftlichen Regeln für Anwendungssysteme umzusetzen (Clark und Wilson 1987, S. 187). Eine grundlegende Annahme im Clark-Wilson Modell ist, dass ein System, welches einen überprüften integren Anfangszustand besitzt, durch eine wohlgeformte, integritätserhaltende Transaktion wieder in einen integren Folgezustand überführt werden kann (Seufert 2001, S. 98). Darauf aufbauend ist eine Menge von Regeln beschrieben, die bei der Durchführung von Operationen zu erfüllen sind. Zur Beschreibung der Regeln wurden folgende Begriffsdefinitionen festgelegt.

3.2.5.1 Clark-Wilson Begriffe

CDI **C**onstraint **D**ata **I**tems (zu schützende integre Datenobjekte): Zu Beginn werden die Datenobjekte identifiziert und gekennzeichnet, für die das Integritätsmodell gilt.

UDI **U**nconstraint **D**ata **I**tems. UDIs sind kritische Komponenten in diesem Integritätsmodell. Nicht alle Daten sind CDIs, weil z. B. über die Tastatur

Informationen, die ein Subjekt eingibt, ins System kommen. CDI und UDI sind disjunkte Mengen.

IVP **I**ntegrity **V**erification **P**rocedures (integritätsverifizierende Prozeduren): Die IVPs bestätigen, dass alle CDIs im System zum Zeitpunkt der Ausführung der IVPs konform mit den Integritätsspezifikationen sind.

TP **T**ransformation **P**rocedures (Tranformationsprozeduren): TPs sind wohl definierte Transaktionen und bilden Subjekte ab. Alle TPs erwarten als Parameter CDIs und das System lässt nur zu, dass TPs den Zustand von CDIs verändern. Eine TP stellt sicher, dass ein integres Datenobjekt nach der Ausführung ebenfalls wieder ein integres Datenobjekt ist. Da innerhalb einer TP die Integrität nicht gewährleistet ist, können TPs nur sequenziell nacheinander abgearbeitet werden (Amoroso 1994, S. 148–149; Clark und Wilson 1987, S. 189).

3.2.5.2 Clark-Wilson Regeln

Die Integritätssicherheit im Clark-Wilson Modell unterscheidet zwei Arten von Regeln:

- Die Zertifizierung (Certification) wird von einem Sicherheitsbeauftragten unter Beachtung der Sicherheitsrichtlinien durchgeführt (Außensicht). Die Zertifizierung (C) gewährleistet gültige Systemzustände durch Definition von Vor- bzw. Nachbedingungen bei der Ausführung von Prozeduren. Zertifizierungsregeln beziehen sich auf anwendungsspezifische Integritätsdefinitionen.
- Die Durchführungsregeln (Enforcement) beschreiben die Innensicht und werden vom System ausgeführt. Durchführungsregeln (E) beschränken die Handlungsfreiheit der Prozeduren, damit die Systemintegrität nicht verletzt werden kann (Clark und Wilson 1987, S. 189; Seufert 2001, S. 99). Durchführungsregeln beziehen sich auf anwendungsunabhängige Sicherheitsfunktionen.

Die ersten drei Zertifizierungsregeln bilden einen Rahmen und spezifizieren die interne Konsistenz von CDIs:

C1 Alle **IVP**s müssen sicherstellen, dass alle **CDI**s während der Ausführung des **IVP** in einem gültigen Zustand sind.

C2 Alle **TP**s müssen zertifiziert sein, um gültig zu sein. Sie überführen **CDI**s, die vor der Ausführung in einem integren Anfangszustand sind, in einen integren Endzustand. Der Sicherheitsbeauftragte definiert eine Relation: (TP_i, (CDI_a, CDI_b, CDI_c, …)). Damit ist die spezifische Menge an Argumenten definiert, für die ein **TP** zertifiziert ist.

E1 Das System verwaltet die Liste der Relationen aus **C2**. Es hat sicherzustellen, dass alle **TP**s nur auf die **CDI** angewandt werden, die sich in der Liste der definierten Relationen aus **C2** befinden (Clark und Wilson 1987, S. 189–190).

Die folgenden weiteren sechs Regeln stellen zum einen die interne und externe Konsistenz und zum anderen die Aufgabentrennung sicher:

E2 Das System verwaltet eine Liste von Relationen, der Form (UserID: (TP_i, (CDI_a, CDI_b, CDI_c, ...)). Damit werden einem Subjekt **TP**s, mit den ihm erlaubten Datenobjekten, zugeordnet. Das System hat sicherzustellen, dass ein Subjekt nur **TP**s, die in der zugehörigen Relation stehen, aufrufen kann. **E2** schränkt die Relation aus **E1** weiter ein.

C3 Die Liste der Relationen aus **E2** muss zertifiziert werden, damit sie den Anforderungen der Aufgabentrennung genügen.

E3 Es ist eine eindeutige **UserID** erforderlich. Eine vorgelagerte Authentifizierung ermittelt die Identität eines jeden Subjekts, das eine **TP** ausführen will.

C4 Alle **TP**s müssen ihre Transaktionen protokollieren. Es müssen alle Informationen protokolliert werden, um die durchgeführten Operationen rekonstruieren zu können. Der Systemadministrator muss diese **TP**s zertifizieren.

C5 **TP**s, die als Eingabeparameter **UDI**s übernehmen und diese dann zu einem **CDI** modifizieren, ein neues **CDI** erzeugen oder - falls dies nicht möglich ist - das **UDI** zurückweisen, müssen ebenfalls zertifiziert werden.

E4 Sicherheitsadministratoren, die autorisiert sind, **CDI**s bzw. **TP**s zu verifizieren, dürfen keine weiteren Zugriffsrechte auf diese erhalten (Clark und Wilson 1987, S. 190–191).

3.2.5.3 Einordnung und Diskussion

Nach der Einordnung in die Klassifikation (siehe Kapitel 3.1.3) findet eine allgemeine Beurteilung des Modells statt:

- Das Clark-Wilson Modell wurde für den kommerziellen Bereich entwickelt, um die Integrität der Daten zu schützen. Da alle Prozeduren überprüft werden müssen, eignet sich das Modell nur für gut überprüfbare Transaktionen, vornehmlich bei einer individuellen Anwendungsentwicklung (Seufert 2001, S. 101).
- Die unterstützte Zugriffskontrollstrategie kann als systemweit eingestuft werden.
- Der Kontrollbereich umfasst die Zugriffskontrolle.
- Die Zugriffsbeschränkung wird alleine durch das Modell geregelt und hängt nicht vom Zustand oder dem Zugriff des Subjektes ab.
- Die Zugriffsrechte werden im Modell nicht vorgegeben, sondern können objektspezifisch vorgegeben werden.
- Die Granularität der Subjekte und Objekte ist beliebig, aber durch die notwendigen Zertifizierungen kann schnell eine hohe Komplexität erreicht werden.
- Über die Administration selbst wird vom Modell nichts ausgesagt.
- Die Aufgabentrennung wird im Modell berücksichtigt.
- Das Prinzip der minimalen Rechte muss bei der Implementierung berücksichtigt werden, da das Modell dazu keine Aussage macht.

Dieses Modell verwendet Transaktionen (**TP**) und zu schützende Datenobjekte (**CDI**). Damit kann eine Integritätspolitik, die über eine Folge von Transaktionen geht, erfasst werden. Jedoch ist das Clark-Wilson Modell nicht einfach zu implementieren. Zudem hat es den Nachteil, dass verschiedene Abstraktionsebenen vermischt werden. Transaktionen und **CDI**s sind computerbasierende Abstraktionen. Daneben sind Notationen für die Verifizierung, Organisations- und Subjektverteilung erforderlich. Der Vorgang der Zertifizierung ist komplex und muss bei jeder Programmänderung wieder erfolgen. Deshalb sollten die Zertifizierungsregeln möglichst minimiert werden (Clark und Wilson 1987, S. 191). Es handelt sich dabei eher um ein Entwurfsmuster als um ein Zugriffskontrollmodell (Seufert 2001, S. 101; Amoroso 1994, S. 154).

3.2.6 Chinese-Wall-Modell

Das Chinese-Wall-Modell ist ebenfalls ein Zugriffskontrollmodell für den kommerziellen Bereich, insbesondere für Banken und Beratungsunternehmen (Brewer und Nash 1989). Dieses Modell wurde entwickelt, um zu verhindern, dass Berater mit vertraulichen Informationen aus einem Unternehmen einen Konkurrenten beraten oder durch Insiderwissen persönliche Vorteile erhalten können. Einem Berater ist es jedoch gestattet, Unternehmen zu beraten, die zu anderen Branchen gehören und nicht direkt mit dem zu beratenden Unternehmen im Wettbewerb stehen. Die grundlegende Idee des Chinese-Wall-Modells ist zukünftige Zugriffsmöglichkeiten auf Informationen eines Subjektes zu beschränken. Die Basis dazu bilden die Zugriffe, die ein Subjekt in der Vergangenheit durchgeführt hat. Denn dadurch erlangte das Subjekt Informationen, die mit einer zukünftigen Beratung in Konflikt stehen. Damit diese Informationen bei der Entscheidung eines Zugriffs auf Objekte zur Verfügung stehen, werden alle Zugriffe protokolliert (Brewer und Nash 1989, S. 207; Eckert 2012, S. 280).

Die Grundlage der Zugriffskontrolle im Chinese-Wall-Modell ist eine Zugriffsmatrix $\mathbf{M_t}$ (siehe Kapitel 3.2.1). Diese entscheidet grundsätzlich, ob ein Zugriffsrecht besteht. In einer zweiten Matrix, der Historienmatrix, werden die bereits getätigten Zugriffe der Subjekte auf Objekte gespeichert. Mit dieser wird über systemweite Regeln des Chinese-Wall-Modells entschieden, ob der Zugriff tatsächlich erlaubt ist. Es wird in diesem Modell eine Trennlinie, die sog. Chinese Wall gezogen. Diese liegt zwischen Informationen, die ohne Interessenkonflikt aufgerufen werden können und solchen, die zu einem Interessenkonflikt führen. Diese Trennlinie ist dynamisch. Zu Beginn der Tätigkeit besteht keine Einschränkung für einen Berater. Mit dem ersten Aufruf eines Klienten aus einer bestimmten Branche sind alle anderen Klienten derselben Branche gesperrt (Brewer und Nash 1989, S. 207; Eckert 2012, S. 280–281; Seufert 2001, S. 103).

3.2.6.1 Das formale Chinese-Wall-Modell

Um eine Autorisierung durchführen zu können, werden die Objekte in neutrale Objekte und in Objekte, die Interessenkonfliktklassen zugeordnet werden, unterschieden.

Die Zuordnung der Objekte zu Unternehmen und Interessenkonfliktklassen geschieht wie **Abb. 3-9** veranschaulicht in drei hierarchischen Stufen:

- Objekte: Auf der untersten Ebene sind Objekte, die individuell gespeicherte Information enthalten, auf die Zugriff benötigt wird.
- Unternehmen: Die mittlere Ebene gruppiert die Objekte, zu welchen Unternehmen $\mathbf{y} \in \mathbf{Y}$ die Objekte gehören.
- Konfliktklasse: Die oberste Ebene bildet die Gruppe der Interessenkonfliktklasse. Hier werden Unternehmen zusammengefasst, die in der derselben Branche miteinander im Wettbewerb stehen (Brewer und Nash 1989, S. 207).

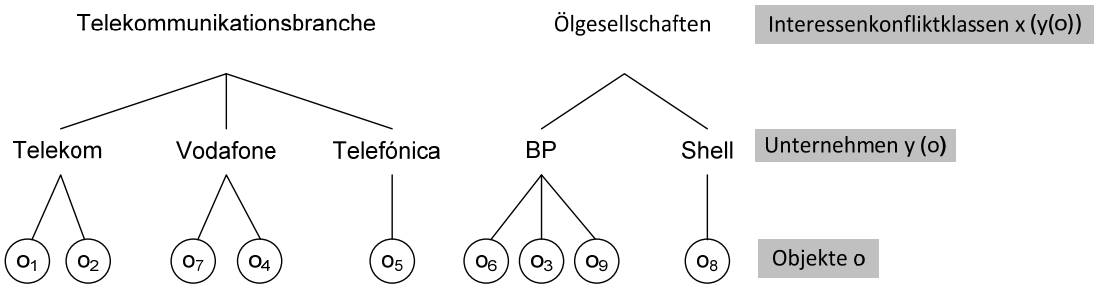

Abb. 3-9 Objekt-Baum im Chinese-Wall-Modell nach (Eckert 2012, S. 281)

Es existieren folgende Komponenten im Chinese-Wall-Modell:

- Eine Menge von Objekten **O**.
- Eine Menge von Subjekten **S**.
- Eine Menge von Sicherheitskennzeichen **K = (x, y)**. Jedes dieser $\mathbf{o} \in \mathbf{O}$ erhält ein solches Kennzeichen $\mathbf{k} \in \mathbf{K}$, wobei **x** die Interessenkonfliktklassen repräsentiert und **y** das Unternehmen beschreibt, zu dem **o** gehört. Für neutrale oder öffentliche Unternehmen und Konfliktklassen für die keine Einschränkung gilt, wird die Notation $\mathbf{x_0}$ bzw. $\mathbf{y_0}$ eingeführt.
- Eine Zugriffsmatrix $\mathbf{M_t}$.
- Eine Historienmatrix **N (s, o)**, wobei die Zeilen die Subjekte repräsentieren und die Spalten die Objekte. Hat $\mathbf{s_1}$ Zugriff auf $\mathbf{o_1}$ erhalten, wird **true** in die Zelle des Schnittpunktes von $\mathbf{s_1}$ und $\mathbf{o_1}$ eingetragen (Brewer und Nash 1989, S. 206–208; Seufert 2001, S. 103–105).

Der Zugriff eines Subjektes $\mathbf{s_m}$ zu einem Objekt $\mathbf{o_n}$ wird gewährt, wenn für alle Einträge in $\mathbf{M_t\ (s_m, o) = true}$ lauten und

$$\mathbf{((x(y(o_n)) \neq x(y(o)) \vee (y(o_n) = y(o)) \vee (y(o_n) \in y_0 \wedge x(y(o_n)) \in x_0).}$$

Beim ersten Zugriff eines Subjektes besteht die Freiheit ein beliebiges Objekt aufzurufen. Nachdem die erste Wahl getroffen wurde, wird eine **Chinese Wall** für dieses Subjekt um diesen Datenbestand herum erzeugt (Seufert 2001, S. 104).

Abb. 3-10 verdeutlicht, wie diese Mauer gezogen wird. Das Subjekt s_m hatte Zugriff auf o_7, damit wird eine Mauer um Vodafone gezogen. Für das Subjekt s_m werden nun die Zugriffe auf Objekte der Firmen Telekom und Telefónica aufgrund der Chinese Wall nicht mehr gestattet. Ölgesellschaften hingegen können vom Subjekt s_m noch beliebig und ohne Einschränkung aufgerufen werden.

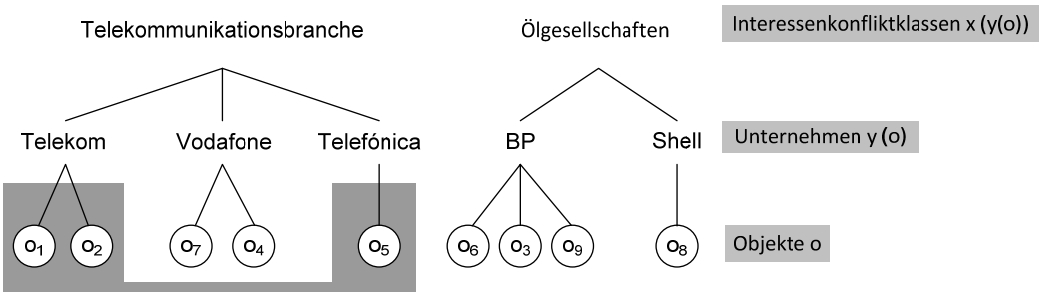

Abb. 3-10 Maueraufbau nach Zugriff auf Objekt o_7 nach " (Eckert 2012, S. 284)

3.2.6.2 Einordnung und Diskussion

Nachstehende Zusammenfassung ordnet das Chinese-Wall-Modell in die Klassifikation (siehe Kapitel 3.1.3) ein:

- Das Chinese-Wall-Modell unterstützt die drei Sicherheitsziele Vertraulichkeit, Integrität und Verbindlichkeit. Die zugrunde liegende Zugriffskontrollmatrix stellt die Erfüllung der Integrität sicher, die Geschäftsbedingungen werden durch die Implementierung der Chinese-Wall umgesetzt und erfüllt die Vertraulichkeit. Indem jeder Zugriff in der Historienmatrix protokolliert wird, wird die Verbindlichkeit erreicht.
- Aufbauend auf der benutzerbestimmten Zugriffskontrollstrategie wird durch die Chinese-Wall eine systemweite Zugriffskontrollstrategie umgesetzt.
- Obwohl das Chinese-Wall-Modell den Informationsfluss zwischen Objekten beschränkt, gehört es zur Klasse der Zugriffskontrollmodelle, da der Informationsfluss für Subjekte und nicht zwischen Objekten beschränkt wird.
- Die Zugriffsrechtsbeschränkung ist zustandsabhängig. Allein das Zugriffsrecht aus der Zugriffsmatrix reicht für die Erlaubnis eines Zugriffes nicht aus, sondern es werden zusätzlich die Historienmatrix sowie Interessenkonfliktklassen ausgewertet.

- Die Zugriffsrechte sind fest vorgegeben.
- Obwohl in dem Modell selbst nur von Dateien und Subjekten gesprochen wird, können für Objekte und Subjekte eine beliebige Granularität für die Implementierung gewählt werden.
- Die Administration erfolgt zentral.
- Das Chinese-Wall-Modell verwendet das Konzept der Aufgabentrennung bei der Zugriffskontrolle (Anderson 2001, S. 166). Es basiert auf der Zugriffsmatrix und erweitert diese um kontextabhängige Bedingungen und Aufgabentrennung.
- Das Prinzip der minimalen Rechte kann je nach Granularität der Objekte und Subjekte sowie der Definition der Interessenkonfliktklassen gewährleistet werden.

3.2.7 Rollenbasiertes Zugriffskontrollmodell (RBAC)

Das rollenbasierte Zugriffskontrollmodell (engl. Role Based Access Control, kurz RBAC) wurde 1992 am National Institute of Standards and Technology (NIST) vorgestellt (Ferraiolo und Kuhn 1992). In den darauf folgenden Jahren wurden u. a. verschiedene Varianten des RBAC-Modells publiziert (Ferraiolo et al. 1995; Giuri und Iglio 1996; Nyanchama und Osborn 1994; Sandhu et al. 1996). Bis heute werden neue Eigenschaften, Erweiterungen und Anwendungen für RBAC-Modelle entwickelt und in jährlichen Workshops diskutiert. Als Konsens aus diesem Prozess bildete sich die Notwendigkeit einer Standardisierung heraus. Nach dem Standardisierungsprozess wurde der von NIST (Ferraiolo et al. 2001) publizierte Entwurf im Februar 2004 als ANSI-Standard verabschiedet (ANSI INCITS 359-2004 2004).

Kurz lässt sich das rollenbasierte Zugriffskontrollmodell wie folgt beschreiben: In RBAC werden Zugriffsrechte an Rollen gebunden, Subjekten werden Rollen zugewiesen. Dadurch erwirbt das Subjekt die den Rollen zugeordneten Zugriffsrechte. RBAC reguliert die Zugriffsrechte von Subjekten über Rollen, damit diese ihre Aufgaben erfüllen können (Ahn und Sandhu 2000, S. 208).

3.2.7.1 Komponenten des RBAC-Modells

Der ANSI-Standard umfasst die Spezifikation eines RBAC-Referenzmodells und der dazugehörigen Funktionen. Das Referenzmodell umfasst ein Kernmodell, Rollenhie-

rarchien sowie statische und dynamische Aufgabentrennung. Für jede dieser Komponenten werden obligatorische und fakultative Funktionen definiert, die eine Implementierung des RBAC-Referenzmodells beinhalten sollte. Die Beschreibung des Referenzmodells wird wie folgt unterteilt in:

- Kernmodell,
- Rollenhierarchie,
- Aufgabentrennung (ANSI INCITS 359-2004 2004, S. 2).

Ein Zugriffskontrollsystem auf Basis des Referenzmodells muss mindestens das Kernmodell enthalten. Die weiteren Komponenten wie Rollenhierarchie, statische und dynamische Aufgabentrennung basieren auf dem Kernmodell und sind voneinander unabhängig und somit orthogonal verwendbar (ANSI INCITS 359-2004 2004; Ferraiolo et al. 2001).

3.2.7.2 Kernmodell des RBAC-Modells

Die Elemente und Beziehungen des Kernmodells sind in **Abb. 3-11** dargestellt. Das Kernmodell umfasst folgende Basiselemente:

- Subjekt (S),
- Rolle (Ro),
- Objekt (O),
- Operator (Op),
- Zugriffsrecht (Z)
- und Sitzung (Si).

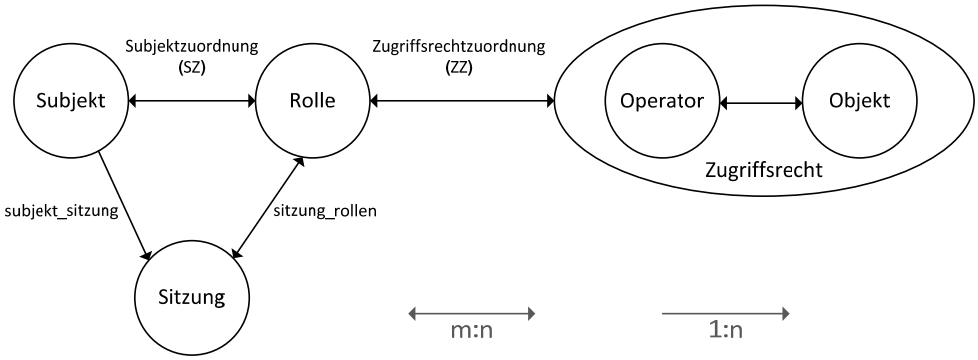

Abb. 3-11 Kernmodell der rollenbasierten Zugriffskontrolle nach (Ferraiolo et al. 2001, S. 232)

Subjekte repräsentieren im Kernmodell personelle Aufgabenträger. Eine **Rolle** ist eine Funktion innerhalb des Kontextes einer Organisation, die Autorität und Verantwortung an das zugeordnete Subjekt überträgt (ANSI INCITS 359-2004 2004, S. 3).

Rollen bündeln im Kernmodell Zugriffsrechte. Subjekten werden danach Rollen zugeordnet. **Zugriffsrechte** sind Genehmigungen, einen **Operator** auf einem oder mehreren von RBAC geschützten **Objekten (O)** auszuführen (ANSI INCITS 359-2004 2004, S. 3). Ein RBAC-Zugriffskontrollmodell funktioniert nach dem Erlaubnisprinzip (siehe Kapitel 2.4.3). Objekte und Operatoren sind abhängig von dem jeweiligen zu schützenden Anwendungssystem (ANSI INCITS 359-2004 2004, S. 3).

> *In einem Dateisystem sind diese Objekte z. B. Dateien oder Verzeichnisse, auf die mit Operatoren wie **schreiben** oder **lesen** zugegriffen wird. Innerhalb eines Datenbankverwaltungssystems sind es Tabellen und Spalten und die Operatoren sind **einfügen**, **ändern**, **löschen**, **selektieren**. In einem Anwendungssystem, z. B. Prüfungsverwaltungssystem, kann ein Objekt u. a. ein Datenblatt eines Studierenden sein und die Operatoren, die darauf angewendet werden können sind: **bearbeiten**, **drucken**, **lesen**.*

Das zentrale Konzept von RBAC ist die Rolle. Dabei formuliert die Rolle die umzusetzende Zugriffskontrollstrategie. **Abb. 3-11** zeigt die Beziehungen **Subjektzuordnung (SZ)** und **Zugriffsrechtzuordnung (ZZ)**. Zwischen Subjekten und Rollen und zwischen Rollen und Zugriffsrechten besteht eine m:n Beziehung. Einem Subjekt kann mehr als eine Rolle zugeordnet werden und eine Rolle kann mehreren Subjekten zugeordnet sein. Dies liefert eine hohe Flexibilität bei der Zuordnung von Zugriffsrechten zu Rollen und Subjekten zu Rollen (ANSI INCITS 359-2004 2004, S. 4).

Eine **Sitzung** ist eine Verbindung zwischen einem Subjekt und der aktivierten Teilmenge von Rollen, die einem Subjekt zugeordnet sind. Jede Sitzung bildet genau ein Subjekt mit allen aktivierten Rollen ab. Ein Subjekt kann beliebig viele Sitzungen eröffnen. Die Funktion **subjekt_sitzung** gibt die mit einem Subjekt verbundene Sitzung aus, während die Funktion **sitzung_rollen** alle aktivierten Rollen in einer Sitzung zurückgibt. Die daraus verfügbaren Zugriffsrechte sind die von den aktivierten Rollen über alle Sitzungen hinweg aktivierten Zugriffsrechte (ANSI INCITS 359-2004 2004, S. 4).

Definition 3-2 Zusammenfassung des Kernmodells des RBAC nach (Ferraiolo et al. 2001, S. 234)

- Subjekt (S), Rolle (Ro), Objekt (O) und Operator (Op) sowie Sitzung (Si)
- $Z = 2^{O} \times {}^{Op}$, eine Menge von Zugriffsrechten.

- $ZZ \subseteq Z \times Ro$, eine m:n Zuordnungsrelation von Zugriffsrechten zu Rollen. Über ein Tupel (z, ro) wird mit einer Rolle $ro \in RO$ ein benötigtes Zugriffsrecht $z \in Z$ assoziiert.
- Die Funktion zugeordnete_zugriffsrechte (ro:Ro) $\rightarrow 2^Z$ beschreibt die einer Rolle zugeordneten Menge an Zugriffsrechten: zugeordnete_zugriffsrechte (ro) = $\{z \in Z \mid (z, ro) \in ZZ\}$
- Op (z:Z) $\rightarrow$ $\{op \subseteq Op\}$ ist die Zugriffsrecht-Operatoren Zuordnung, die eine Menge an Operatoren ausgibt, die einem Zugriffsrecht zugeordnet sind.
- O (z:Z) $\rightarrow$ $\{o \subseteq O\}$ ist die Zugriffsrecht-Objekt Zuordnung, die eine Menge an Objekten ausgibt, die einem Zugriffsrecht zugeordnet sind.
- $SZ \subseteq S \times Ro$, eine m:n Zuordnungsrelation von Subjekt zu Rollen. Ein Tupel (s, ro) $\in SZ$ legt die Rollenmitgliedschaft eines Subjektes $s \in S$ in einer Rolle $ro \in Ro$ und den damit verbundenen Zugriffsrechten fest.
- Die Funktion zugeordnete_subjekte: (ro:Ro) $\rightarrow 2^S$ mit zugeordnete_subjekte (r) = $\{s \in S \mid (s, ro) \in SZ\}$ bestimmt die Menge der einer Rolle zugeordneten Subjekte.
- subjekt_sitzung S $\rightarrow 2^{Si}$, ist die Abbildung von einem Subjekt $s \in S$ auf eine Menge von Sitzungen $si \in Si$.
- sitzung_subjekt (si:Si) $\rightarrow 2^S$, ist die Abbildung von einer Sitzung $si \in Si$ auf das korrespondierende Subjekt $s \in S$.
- sitzung_rollen Si $\rightarrow 2^{Ro}$, die Abbildung von einer Sitzung $si \in Si$ auf eine Menge von Rollen. sitzung_rollen (si) $\subseteq$ $\{ro \in Ro \mid$ (sitzung_subjekt (si), ro) $\in ZZ\}$ bezeichnet die aktivierten Rollen einer Sitzung.
- verfügbare_sitzung_zugriffsrechte (si:Si) $\rightarrow 2^Z$, in einer Sitzung $si \in Si$ verfügbaren Zugriffsrechte eines Subjektes.
  verfügbare_sitzung_zugriffsrechte(si) = $\{ro \in Ro \mid \exists$ sitzung_rollen (si) $\wedge \exists$ zugeordnete_zugriffsrechte (ro)$\}$.

3.2.7.3 Rollenhierarchie

Die zweite Komponente des RBAC-Modells erweitert das Kernmodell um die **Rollenhierarchie (RH)** (ANSI INCITS 359-2004 2004, S. 5). **Abb. 3-12** zeigt die Einbindung der Rollenhierarchien in das Kernmodell. Die Rollenhierarchie ist eine

Möglichkeit, um organisatorische Beziehungen abzubilden (Ferraiolo et al. 2001, S. 234).

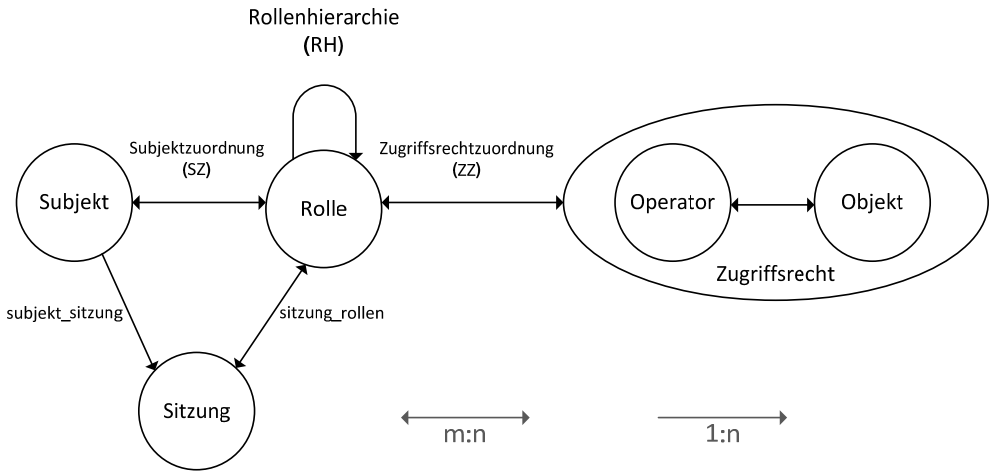

Abb. 3-12 RBAC-Modell nach (Ferraiolo et al. 2001, S. 235)

Rollenhierarchien definieren eine Vererbungsbeziehung zwischen Rollen und werden als Zugriffsrechtsvererbung oder Vererbung der Subjektmitgliedschaft beschrieben (ANSI INCITS 359-2004 2004, S. 5). Zugriffsrechtsvererbung beschreibt, eine Rolle ro_1 „erbt von" Rolle ro_2, wenn alle Zugriffsrechte von ro_2 auch Zugriffsrechte von ro_1 sind. Eine Vererbung der Subjektmitgliedschaft beschreibt: Rolle ro_1 „beinhaltet" Rolle ro_2, wenn alle Subjekte, die für ro_1 autorisiert sind, auch für ro_2 autorisiert sind. Eine Subjektmitgliedschaft unterstellt, dass ein Subjekt der Rolle ro_1 mindestens alle Zugriffsrechte von ro_2 hat. Umgekehrt sagt die Zugriffsrechtsvererbung von ro_1 und ro_2 nichts über die Subjektmitgliedschaft aus (Ferraiolo et al. 2001, S. 234). Neben der in den Standard aufgenommenen Zugriffsrechtsvererbung bzw. Vererbung der Subjektmitgliedschaft finden sich weitere Interpretationen von Rollenhierarchien in der Literatur[30].

Der ANSI-Standard beschreibt eine allgemeine und eine beschränkte Rollenhierarchie. Eine allgemeine Rollenhierarchie ist eine beliebige partielle Ordnung. Sie unterstützt Mehrfachvererbung, um Zugriffsrechte bzw. Subjektmitgliedschaften von zwei oder mehr Quellen zu erben. Eine Mehrfachvererbung hat zwei Vorteile bei der Umsetzung von Hierarchieeigenschaften:

[30] Zum weitergehenden Studium wird auf Kapitel 4.3 und Kuhn (1998); Nyanchama und Osborn (1999); Sandhu (1998) verwiesen.

- Eine Rolle kann aus mehreren Vorgängerrollen entworfen werden, indem Rollen und Beziehungen definiert werden, die charakteristisch für die zu repräsentierende Organisationsstruktur und Aufgabenverteilung sind.
- Es entsteht durch Mehrfachvererbung eine Gleichbehandlung von Subjektzuordnungen (SZ) zu Rollen im Kernmodell und Rollen-Rollen-Vererbungsrelationen (RH). Sowohl bei Subjektordnung als auch bei der Vererbung der Zugriffsrechte und der Subjektmitgliedschaft kann dieselbe „$\geq$“ Relation verwendet werden (ANSI INCITS 359-2004 2004, S. 6–7).

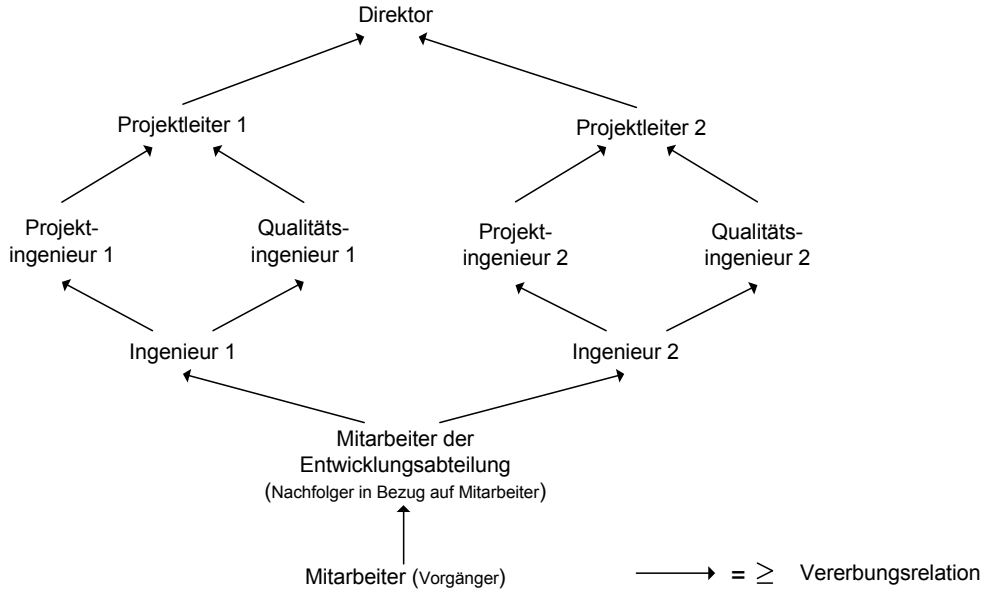

Abb. 3-13 Allgemeine Rollenhierarchie nach (Ferraiolo et al. 2001, S. 236)

Abb. 3-13 beschreibt als Beispiel einer allgemeinen Rollenhierarchie eine Entwicklungsabteilung. Die Wurzel der Hierarchie bildet die Rolle **Mitarbeiter**. Dieser Rolle sind alle Zugriffsrechte zugeordnet, die alle Mitarbeiter unabhängig von ihrer Position benötigen. Ein Projektleiter erhält Zugriffsrechte sowohl von der Rolle **Projektingenieur** als auch von der Rolle **Qualitätsingenieur** eines Projektes.

Definition 3-3 Allgemeine Rollenhierarchie nach (Ferraiolo et al. 2001, S. 235)

– $RH \subseteq Ro \times Ro$ ist eine partielle Ordnung auf Rollen, die Vererbungsrelation genannt wird, geschrieben „$\geq$“, wobei $ro_1 \geq ro_2$ nur gilt, wenn alle Zugriffsrechte von ro_2 auch die Zugriffsrechte von ro_1 oder alle Subjekte von ro_1 auch Subjekte von ro_2 sind.
$ro_1 \geq ro_2 \Rightarrow$ autorisierte_zugriffsrechte(ro_2) $\subseteq$ autorisierte_zugriffsrechte(ro_1)

- autorisierte_subjekte (ro:Ro) $\rightarrow 2^S$, die Zuordnung von Rollen auf eine Menge von Subjekten in Berücksichtigung der Rollenhierarchie.
  autorisierte_subjekte (ro) = {s $\in$ S | ro' $\geq$ ro, (s, ro' $\in$ SZ}
- autorisierte_zugriffsrechte (ro:Ro) $\rightarrow 2^Z$, die Zuordnung von Rollen auf eine Menge von Zugriffsrechten unter Berücksichtigung der Rollenhierarchie.
  autorisierte_zugriffsrechte (ro) = {z $\in$ Z | ro' $\geq$ ro, (z, ro' $\in$ ZZ}

Eine Rolle kann bei der beschränkten Rollenhierarchie einen oder mehrere direkte Nachfahren haben, aber die Rolle ist beschränkt auf einen einzigen unmittelbaren Vorgänger. Die Restriktionen der beschränkten Rollenhierarchien führen zu einer einfachen Baumstruktur. Jedoch können innerhalb einer beschränkten Rollenhierarchie Subjekte von mehr als einer Rolle erben, zu sehen am Beispiel Frank in **Abb. 3-14**. Obwohl die beschränkte Rollenhierarchie keine Mehrfachvererbung unterstützt, liefert sie administrative Vorteile gegenüber dem RBAC-Kernmodell (ANSI INCITS 359-2004 2004, S. 7).

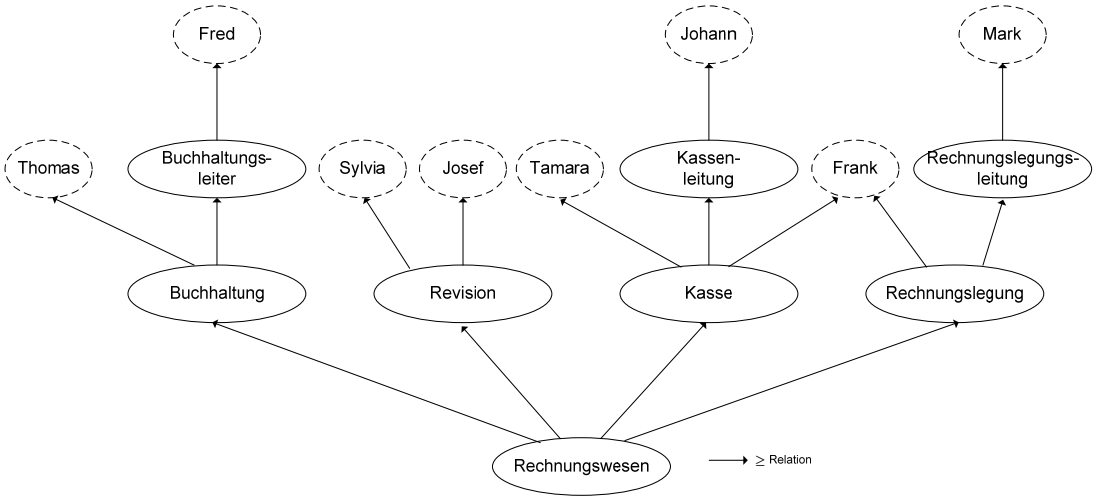

Abb. 3-14 Beschränkte Rollenhierarchie am Beispiel des Rechnungswesens (Ferraiolo et al. 2001, S. 237)

Definition 3-4 Beschränkte Rollenhierarchie

- Allgemeine Rollenhierarchie mit der folgenden Einschränkung: $ro_1 \geq ro_2$
  $\forall$ ro, ro_1, $ro_2 \in$ Ro, ro $\geq ro_1 \wedge$ ro $\geq ro_2 \Rightarrow ro_1 = ro_2$

Abb. 3-14 illustriert eine beschränkte Rollenhierarchie am Beispiel von Rollen im Rechnungswesen. Im Rollengraph werden Subjekte als Ellipsen mit den gestrichelten Linien dargestellt und Rollen durch einfache Ellipsen symbolisiert.

Johann ist der Rolle Kassenleitung zugeordnet und damit auch autorisiert für die Rollen Kasse und Rechnungswesen. Johanns Zugriffsrechte setzen sich zusammen aus den Zugriffsrechten für Kassenleitung, Kasse und Rechnungswesen.

3.2.7.4 Aufgabentrennung im RBAC

Die Aufgabentrennung[31] (siehe Kapitel 2.4.3) kann dem RBAC-Kernmodell als zusätzliche Komponente orthogonal zur Rollenhierarchie hinzugefügt werden. Es gibt zwei Arten von Aufgabentrennung:

- eine statische (SAT) und
- eine dynamische Aufgabentrennung (DAT).

Statische Aufgabentrennung

Interessenskonflikte in einem rollenbasierten Autorisierungssystem können entstehen, wenn ein Subjekt Zugriffsrechte erlangt, die mit konfliktären Rollen verbunden sind. Um die SAT modellieren zu können, stellt das RBAC-Modell das Konzept der sich ausschließenden Rollen, wie **Abb. 3-15** zeigt, zur Verfügung. Bei einer statischen Aufgabentrennung wird bereits bei der Subjektzuordnung überprüft, dass keine sich ausschließenden Rollen demselben Subjekt zugeordnet werden. Da diese Art der Beschränkung auf der administrativen Ebene angesiedelt ist, kann damit eine übergeordnete systemweite organisatorische Aufgabentrennung durchgesetzt werden (ANSI INCITS 359-2004 2004, S. 8).

Die Überwachung der Aufgabentrennung zwischen genau zwei Rollen wird im ANSI-Standard als zu streng angesehen, deshalb wird die Funktion der Aufgabentrennung mit zwei Argumenten versehen, einer Menge von mindestens zwei zuordenbaren Rollen und der Angabe einer Kardinalität. Wird diese Kardinalität bei der Subjektzuordnung aus der Menge der möglichen Rollen überschritten, kommt es zu einer Verletzung der Aufgabentrennung (ANSI INCITS 359-2004 2004, S. 8). Zusätzliche Aspekte der Aufgabentrennung[32] im RBAC sind in Kapitel 4.2 beschrieben.

[31] Weitere Beschreibungen und Diskussionen zur Aufgabentrennung finden sind in Brewer und Nash (1989); Clark und Wilson (1987); Anderson (2001, S. 166–199).

[32] Die Umsetzung der Aufgabentrennung im RBAC wird in zahlreichen Veröffentlichungen beschrieben Ferraiolo et al. (1995); Kuhn (1997); Gligor et al. (1998); Giuri und Iglio (1997).

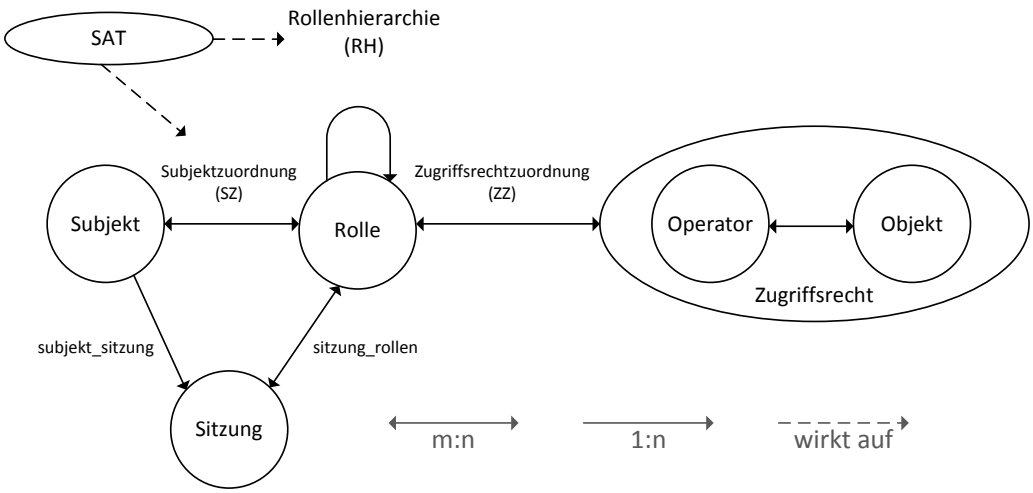

Abb. 3-15 Statische Aufgabentrennung im RBAC nach (ANSI INCITS 359-2004 2004, S. 9)

Definition 3-5 Statische Aufgabentrennung (SAT) (ANSI INCITS 359-2004 2004, S. 10)

- SAT $\subseteq (2^{RO} \times N)$ ist eine Sammlung von Tupel (rom, n) in statischer Aufgabentrennung (SAT), eine Menge von Rollen (rom), eine Teilmenge (t) von Rollen aus der Menge rom, und n ist eine natürliche Zahl ≥ 2, mit der Eigenschaft, dass kein Subjekt zu n oder mehr Rollen aus der Menge rom in jedem (rom, n) $\in$ SAT zugeordnet werden kann.

  $$\forall \text{ (rom, n) } \in \text{ SAT, } \forall \text{ t} \subseteq \text{ rom : } |\text{t}| \geq \text{ n} \Rightarrow \bigcap_{\text{ro} \in \text{t}} \textit{subjektzuordnung} \text{ (ro)} = \varnothing$$

Eine statische Aufgabentrennung kann ohne oder zusammen mit dem Konzept der Rollenhierarchie existieren. Bei der Anwendung von Rollenhierarchien und statischer Aufgabentrennung ist Sorgfalt bei der Administration erforderlich, um sicherzustellen, dass bei der Vererbung von Rollen nicht die statische Aufgabentrennung verletzt wird. Das Konzept verbietet (siehe **Definition 3-6**) eine Zuordnung innerhalb der Rollenhierarchie, damit keine Verletzung der SAT stattfindet (ANSI INCITS 359-2004 2004, S. 9).

Definition 3-6 Statische Aufgabentrennung bei Rollenhierarchie (ANSI INCITS 359-2004 2004, S. 10)

- Bei einer existierenden Rollenhierarchie wird die statische Aufgabentrennung auf *autorisierte_subjekte* anstatt auf *subjektzuordnung* wie folgt neu definiert.

  $$\forall \text{ (rom, n) } \in \text{ SAT } \forall \text{ t} \subseteq \text{ rom : } |\text{t}| \geq \text{ n} \Rightarrow \bigcap_{\text{ro} \in \text{t}} \text{ autorisierte\_subjekte (ro)} = \varnothing$$

Dynamische Aufgabentrennung (DAT)

DAT begrenzt ebenfalls wie SAT die verfügbaren Zugriffsrechte. Der Kontext einer Überprüfung der DAT ist die Aktivierung von Rollen in einer Sitzung. Diese Komponente limitiert die Verfügbarkeit von Zugriffsrechten, indem eine Bedingung auf die Beziehung **sitzung_rollen**, wie in **Abb. 3-16** dargestellt, definiert wird. Diese Bedingung verhindert, dass ein Subjekt die entsprechenden Rollen und damit die zugeordneten Zugriffsrechte gemeinsam in einer Sitzung aktivieren kann.

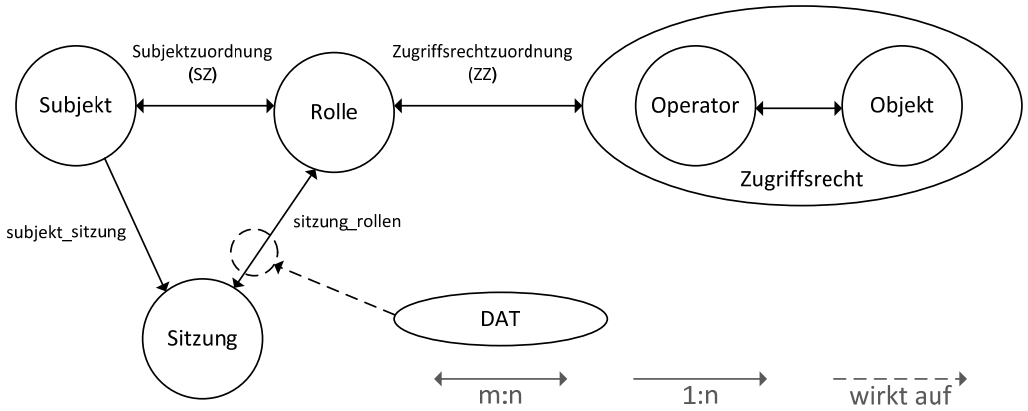

Abb. 3-16 Aufgabentrennung im RBAC nach (ANSI INCITS 359-2004 2004, S. 10)

> *Zum Beispiel kann ein Subjekt für die beiden Rollen Kassier und Kassenrevisor autorisiert sein, wobei es dem Revisor erlaubt ist, die Korrekturen des Kassiers auf einer offenen Barabhebung zu quittieren. Falls ein Subjekt in der Rolle des Kassiers agiert und danach in die Rolle des Kassenrevisors wechseln will, würde das RBAC-Modell fordern, dass das Subjekt die Rolle des Kassiers verlässt und damit den Abschluß der Barabhebung erzwingt, bevor die Rolle des Kassenrevisors übernommen werden kann. Solange es einem Subjekt nicht erlaubt ist, beide Rollen zur selben Zeit anzunehmen, kann kein Interessenskonflikt entstehen.*

Obwohl auch mit einer statischen Aufgabentrennung dieses Ergebnis erzielt wird, bietet die dynamische Aufgabentrennung jedoch eine größere organisatorische Flexibilität (ANSI INCITS 359-2004 2004, S. 10).

Definition 3-7 Dynamische Aufgabentrennung (ANSI INCITS 359-2004 2004, S. 10)

- DAT $\subseteq (2^{RO} \times N)$ ist eine Sammlung von Paaren (rom, n) mit dynamischer Aufgabentrennung, indem jedes rom eine Rollenmenge darstellt und n eine natürliche Zahl ≥ 2 ist, mit der Eigenschaft, dass kein Subjekt n oder mehr Rollen aus der Menge rom in jedem dat $\in$ DAT aktivieren kann.

$\forall$ rom $\in 2^{RO}$, n $\in$ N, $\forall$ (rom, n) $\in$ DAT, n $\geq$ 2 und
$\forall$ si $\in 2^{Si}$, $\forall$ rom $\in 2^{RO}$, $\forall$ rollen_teilmenge $\in 2^{RO}$, $\forall$ n $\in$ N, (rom, n) $\in$ DAT,
rollen_teilmenge $\subseteq$ rom, *rollen_teilmenge* $\subseteq$ *sitzung_rollen* (si) $\Rightarrow$ | *rollen_teilmenge* | < n.

3.2.7.5 Spezifikation der RBAC-Funktionen

Neben den Komponenten des RBAC-Kernmodells, der Rollenhierarchie und der Aufgabentrennung definiert der Standard, Funktionen für jede der Komponenten in den Bereichen Administration, Rechteprüfung und Protokollierung. Dabei wird zwischen obligatorischen und fakultativen Funktionen unterschieden. Es werden die Bedingungen, der Algorithmus und das Ergebnis der Funktionen formal beschrieben. Zum Abschluss der Funktionsspezifikationen zeigt ein Leitfaden, siehe **Abb. 3-17**, wie in Abhängigkeit von den gewählten Teilkomponenten die einzelnen Funktionen zu funktionalen Paketen zusammengefasst werden können[33].

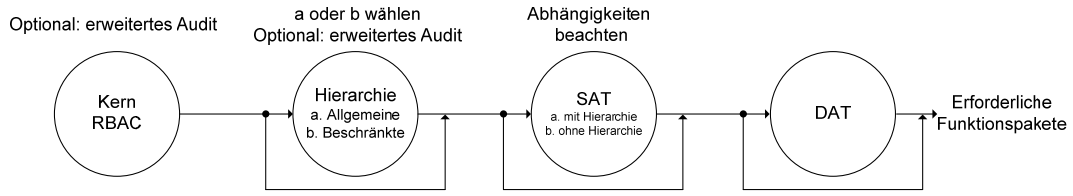

Abb. 3-17 Methode, um funktionale Pakete zu erzeugen nach (ANSI INCITS 359-2004 2004, S. 43)

Das einzige zwingende Funktionspaket sind Funktionen, die sich auf das Kernmodell beziehen. Die anderen Pakete sind abhängig von den jeweilig gewählten Komponenten. Alle Funktionen des Kernmodells und bei Bedarf die des erweiterten Audits, müssen durch die Implementierung übernommen werden. In Abhängigkeit von der gewählten Rollenhierarchie müssen entsprechende Funktionen implementiert werden. Auch hier sind die Funktionen für ein erweitertes Audit optional. Die Funktionen für die statische Aufgabentrennung sind von der Wahl der Hierarchie abhängig. Die Funktionen der dynamischen Aufgabentrennung hängen nur von dieser ab.

[33] Für ein ausführliches Studium der einzelnen Funktionen wird auf den Standard verwiesen ANSI INCITS 359-2004 (2004, S. 11–35)

3.2.7.6 Kritik am Referenzmodell

Nachdem der erste Entwurf des Standards bzw. der Standard veröffentlicht wurde, gab es u. a. folgende Kritik an dessen Umsetzung (Jaeger und Tidswell 2000; Li et al. 2007). Ein Standard sollte ein einheitliches Referenzmodell sein, eine Grundlage für zukünftige Standards bilden und Grundelemente einschließlich einer allgemeinen API definieren, damit Systeme verschiedener Anbieter miteinander kommunizieren können. Die Kritik befindet, dass in den ANSI-Standard nicht mehr aufgenommen wurde als bereits bei den ersten veröffentlichen RBAC-Modellen spezifiziert wurde, obwohl Erweiterungen und Verbesserungen in der Forschung ausführlich diskutiert wurden (Jaeger und Tidswell 2000, S. 65–66).

Folgende Mängel am ANSI-Standard wurden z. B. beschrieben:

- Es werden die vier Komponenten und die Funktionen separat beschrieben. Die Funktionen sollten besser als Interfacebeschreibung bei der jeweiligen Komponente beschrieben werden.
- Bei der Beschreibung wurde wenig Wert darauf gelegt zu erklären, wie Subjekte, Zugriffsrechte und die Bildung der Rollenhierarchie und Aufgabentrennung modelliert werden sollen. Es wurden keine Hinweise gegeben, ob Rollen als organisatorische Rollen, als Aufgaben oder Funktionen innerhalb der Organisation beschrieben werden können. In Kapitel 5 wird dieser Kritikpunkt aufgegriffen und untersucht.
- Über die Implementierung einer Administration von RBAC wurde nichts ausgesagt, obwohl es dazu ausführliche Forschungsarbeiten gab (Jaeger und Tidswell 2000, S. 65–66).

Neben der Kritik am Standard wurden auch Vorschläge unterbreitet, den ANSI-Standard zu verbessern (Li et al. 2007). Das Konzept der Sitzung sollte vom Kernmodell entfernt und in einer eigenen Komponente ausgelagert werden. Sitzungen sind nach Meinung der Autoren nur notwendig bei Datenbankmanagementsystemen. In Anwendungssystemen kann das Prinzip der minimalen Zugriffsrechte auch anderweitig erreicht werden (Li et al. 2007, S. 43). Die Kritik greift nicht, da vor allem in webbasierten Anwendungen ebenfalls Sitzungen existierten. Zudem würden, wenn es keine Sitzungen gäbe, alle Rollen zu jeder Zeit aktiviert werden, was nicht gewollt ist (Ferraiolo et al. 2007, S. 51).

Folgende Kritikpunkte wurden außerdem beschrieben:

- Es sollte nur eine einzige Rolle pro Sitzung aktiviert werden können, um das Prinzip der minimalen Rechte besser gewährleisten zu können (Li et al. 2007, S. 44).
- Der Standard sollte eine explizite Rollen-Dominanzrelation beinhalten und diese sollte auch bei jeder Änderung angepasst werden, damit immer gewährleistet ist, dass die Hierarchie auch beim Löschen und Einfügen von Rollen richtig erhalten bleibt (Li et al. 2007, S. 45–47).
- Die Semantik der Rollenhierarchie sollte klarer beschrieben werden, ob es sich um eine Zugriffsrechtsvererbung, Vererbung der Subjektmitgliedschaft oder Aktivierungshierarchie handelt (Li et al. 2007, S. 47). Kapitel 4.3 greift diesen Aspekt nochmals auf.

3.2.7.7 Einordnung und Diskussion

RBAC ist ein Zugriffskontrollmodell, das Zugriffsrechte nicht direkt Subjekten, sondern Rollen zuordnet. Der ANSI-Standard beschreibt ein Referenzmodell bestehend aus einem Kernmodell und Erweiterungen um Rollenhierarchien und Aufgabentrennung. Außerdem wird eine Protokollierung im Modell festgelegt und es werden Funktionen für die Implementierung definiert. RBAC lässt sich wie folgt in die Klassifikation (siehe Kapitel 3.1.3) einordnen:

- Das von RBAC unterstützte Sachziel ist die Integrität. RBAC unterstützt dabei unternehmensspezifische Sicherheitsstrategien, die sich eng an die Unternehmensstruktur anlehnen. Damit ist eine Gestaltung der Informationssicherheit möglich, die sich an der Unternehmensorganisation orientiert (Schier 1999, S. 164).
- Zu Beginn wurde das RBAC-Modell entweder der systemweiten oder der benutzerbestimmten Zugriffskontrollstrategie zugeordnet. Zentraler Bestandteil von RBAC aber sind Rollen. „Die zu erfüllenden Aufgaben bestimmen“ (Lau und Gerhardt 1994, S. 66) die Zugriffsrechte einer Rolle und dadurch auch die des personellen Aufgabenträgers, der in dieser Rolle agiert (Lau und Gerhardt 1994, S. 66). RBAC setzt damit vom Modell her zunächst eine rollenbasierte Zugriffskontrollstrategie um und unterstützt „eine aufgabenorientierte Modellierung von Berechtigungsvergaben“ (Eckert 2012, S. 296). Untersuchungen ergaben, dass das RBAC-Modell hinsichtlich der Zugriffs-

kontrollstrategie flexibel sein kann (Osborn et al. 2000, S. 86; Essmayr et al. 2004, S. 140). Durch entsprechende Konfiguration von Rollenhierarchien und statischer und dynamischer Aufgabentrennung kann eine systemweite oder benutzerbestimmte Zugriffskontrollstrategie umgesetzt werden (Bauknecht und Holbein 1996, S. 276).

- Der unterstützte Kontrollbereich ist die Zugriffskontrolle.
- Im Referenzmodell ist die Zugriffsbeschränkung zustandsunabhängig, da allein die Aktivierung der Rolle mit dem entsprechenden Zugriffsrecht ausreicht, um das Zugriffsrecht zu erhalten. Es existieren jedoch Erweiterungen, die den Kontext berücksichtigen.
- Die Zugriffsrechte sind objektspezifisch und können an das Zielanwendungssystem angepasst werden.
- Objekte und Subjekte lassen sich in ihrer Granularität in Abhängigkeit vom zu schützenden Anwendungssystem modellieren.
- Die Administration des ANSI-Standards erfolgt zentral. Die Zugriffsrechtszuordnung sowie die Subjektzuordnung werden von einer Systemadministration vorgenommen.
- Die Aufgabentrennung und die Skalierbarkeit sind auf der Modellebene gewährleistet. Durch das Aktivieren einer Teilmenge von Rollen kann eine dynamische Aufgabentrennung gewährleistet werden (Ferraiolo et al. 2007, S. 51).
- Durch die hohe Flexibilität und die Möglichkeit der anwendungsspezifischen Granularität wird das Prinzip der minimalen Rechte unterstützt.

Die Vorteile eines Zugriffskontrollsystems auf Basis des RBAC-Modells sind neben der Möglichkeit Zugriffsrechte in Rollen zu bündeln, in der vereinfachten Administration der Zugriffskontrolle und der Protokollierung zu sehen (Sandhu und Munawer 1999; Kern 2002). Die Vereinfachung der Administration ergibt sich dadurch, dass Rollen in Unternehmen relativ stabil sind, im Gegensatz zu häufigeren Änderungen am Mitarbeiterstamm (Ferraiolo et al. 1999, S. 36). Daneben gibt es auch einen wirtschaftlichen Aspekt, durch das Einführen eines Zugriffskontrollsystems auf Basis des rollenbasierten Zugriffskontrollmodells können die Administrationskosten gesenkt werden (Ferraiolo et al. 1999, S. 36; Herwig und Schlabitz 2004, S. 291; Gallaher et al. 2002).

3.3 Gegenüberstellung und Bewertung der Zugriffsmodelle

Abb. 3-18 zeigt die Gegenüberstellung der betrachtenden Zugriffsmodelle anhand der Klassifikation aus Kapitel 3.1.3, um ein geeignetes Zugriffskontrollmodell für aufgabenbezogene Rollen herauszufinden.

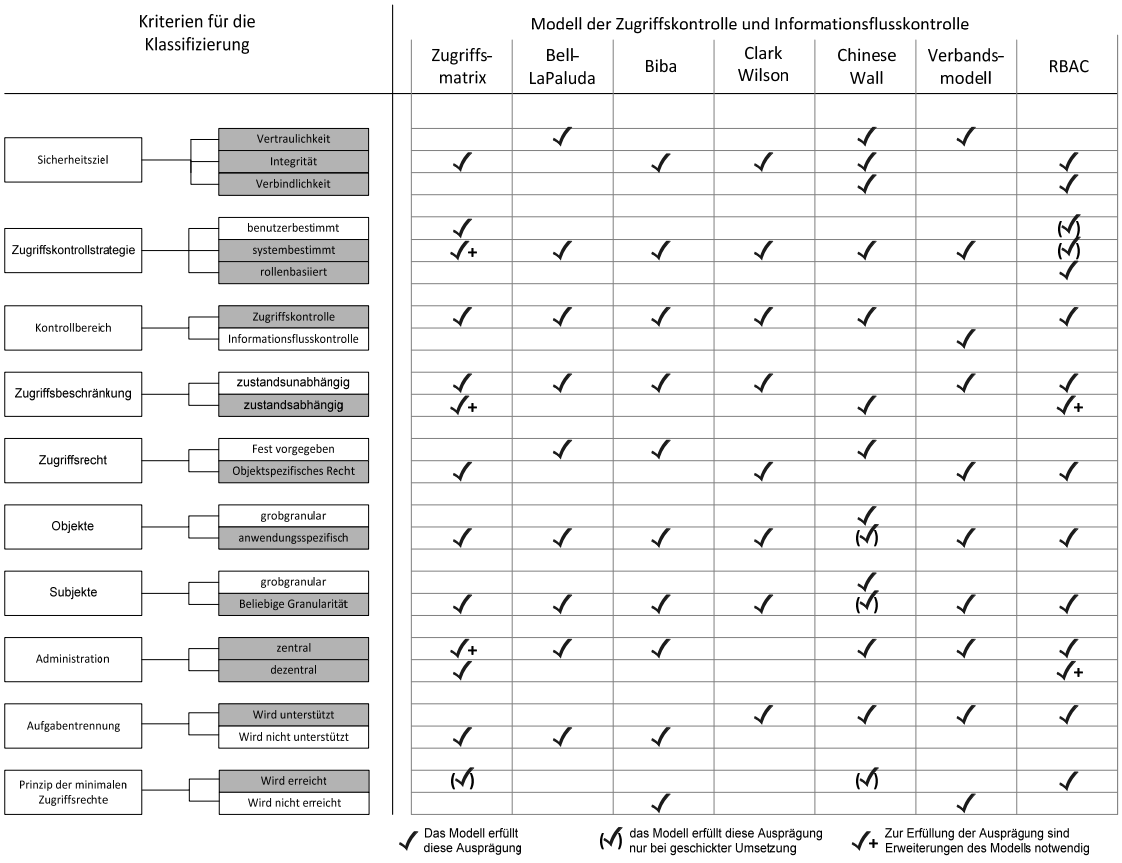

Abb. 3-18 Gegenüberstellung der untersuchten Zugriffskontrollmodelle

Das wichtigste zu erfüllende Sachziel eines Zugriffskontrollmodells im IS ist die Integrität. Damit sind einige der untersuchten Zugriffskontrollmodelle dafür nicht geeignet, da sie andere Sachziele verfolgen: Das Bell-LaPadula-Modell setzt das Sicherheitsziel Vertraulichkeit um. Deshalb hat dieses Modell keine Verbreitung im kommerziellen oder administrativen Bereich gefunden und das reine Zugriffsmatrix-Modell hat sich damals als Modell im kommerziellen Bereich etabliert (Gasser 1988, S. 68). Das Biba-Modell ist von seinem Ansatz her zu restriktiv und unflexibel. Das Verbandsmodell eignet sich nur für die Informationsflusskontrolle und vermengt Geschäftslogik und Zugriffskontrolle. Das Clark-Wilson-Modell ist ausschließlich für eine transaktionsorientierte Zugriffskontrolle geeignet und vermischt die Ebene der maschinellen und personellen Aufgabenträger. Das Chinese-Wall-Modell eignet

sich nicht für allgemeine kommerzielle Anwendungen, da es zu spezifisch für Banken und Berater entwickelt wurde.

Abb. 3-18 zeigt, dass sowohl das Zugriffsmatrix–Modell als auch RBAC geeignet sind, um ein Zugriffskontrollmodell für die Autorisierung von Anwendungssoftware zu entwickeln (Saunders et al. 2001, S. 18). Mit beiden Modellen kann das Sachziel Integrität umgesetzt werden, der Kontrollbereich ist die Zugriffskontrolle und die Zugriffsbeschränkung ist zustandsunabhängig. Es lassen sich die Objekte in ihrer Granularität in Abhängigkeit vom Anwendungssystem sowie Subjekte beliebiger Granularität modellieren. Die Zugriffsrechte beziehen sich auf die zu schützenden Objekte.

RBAC hat jedoch gegenüber der Zugriffsmatrix folgende Vorteile: RBAC stellt im Gegensatz zur Zugriffsmatrix sowohl Konzepte für eine statische als auch dynamische Aufgabentrennung auf Modellebene bereit. Damit kann die Aufgabentrennung (siehe Kapitel 2.3.4) mit dem RBAC-Modell bereits auf Modellebene dargestellt und die Implementierung darauf aufgebaut werden. Die Rechteverwaltung der Zugriffsmatrix ist vom Modell her dezentral, während hier RBAC die Möglichkeit einer zentralen Administration vorsieht. Die Zugriffskontrollstrategie der Zugriffsmatrix ist zunächst benutzerbestimmt, während RBAC eine rollenbasierte Zugriffskontrollstrategie umsetzt.

Autorisierung in verteilten Anwendungen sollte betrachtet werden anhand von Aufgaben und nicht von individuellen Subjekten und Objekten (Thomas und Sandhu 1993, S. 139). Die bis 1994 existierenden Zugriffskontrollmodelle werden auf einer zu niedrigen Abstraktionsebene definiert, als dass sie hilfreich wären, organisatorische Anforderungen und Aspekte der Sicherheitsstrategie zu modellieren. Ein rollenbasiertes Zugriffskontrollmodell kann diese Lücke der Abstraktion schließen (Thomas und Sandhu 1994, S. 66).

„Rollenbasierte Zugriffskontrollmodelle unterstützen eine aufgabenorientierte Modellierung der Berechtigungsprofile“ (Eckert 2012, S. 296). Durch die Bündelung von Zugriffsrechten zu Rollen „sind RBAC-Modelle besonders für Anwendungssysteme, die durch eine sich ändernde Menge von Subjekten und durch eine über lange Zeiträume gleich bleibende Menge von Aufgaben charakterisiert sind, geeignet“ (Eckert 2012, S. 296). Durch die Definition verschiedener Rollen und der Sub-

jektzuordnung je nach Verantwortung in einer Organisation sowie der Rollenaktivierung und Deaktivierung kann eine flexible Autorisierung implementiert werden (Bauknecht und Holbein 1996, S. 277).

Bereits nach Veröffentlichung des Standards wurde Kritik am Referenzmodell geübt und Defizite in Bereichen wie Administration, Delegation und Dynamisierung festgestellt. RBAC kann erweitert werden um Konzepte wie Delegation, negative Rechte, Kontextabhängigkeit der Rollen, Rollenzuordnung durch Regeln oder Parametrisierung von Rollen (Al-Kahtani und Sandhu 2002; Barka und Sandhu 2000a; Ge und Osborn 2004; Hagström et al. 2001; Kern und Walhorn 2005; Zhang et al. 2003b). In Kapitel 4 werden zum einen weitergehende Untersuchungen in Bezug auf Aufgabentrennung und Rollenhierarchie vorgenommen und zum anderen ausgewählte Konzepte für Erweiterungen analysiert.

Das zentrale Element des rollenbasierte Zugriffskontrollmodells ist die Rolle. Um die Entscheidung für ein rollenbasiertes Zugriffskontrollmodell zu untermauern, wird in Kapitel 5 das Konzept der Rolle untersucht, die Beziehung zwischen Aufgaben, Aufgabenträger und Rolle herausgearbeitet und in das Konzept des betrieblichen Informationssystem integriert.

4 Untersuchung ausgewählter Konzepte im RBAC-Modell

Das im vorherigen Kapitel ausgewählte Referenzmodell RBAC wird nachstehend ausführlich analysiert und dabei die geübte Kritik aufgegriffen: Es werden zunächst die Unterschiede zwischen dem Rollen- und Gruppenkonzept (Anderson 2001, S. 54–55; Barkley 1997, S. 130; Gebel 2003, S. 27; Seufert 2001, S. 54-56,112-114), ausgewählte Aspekte der Aufgabentrennung (Gligor et al. 1998; Simon und Zurko 1997) und Rollenhierarchie (Ferraiolo et al. 2003, S. 75–82; Moffett 1998) betrachtet. Zusätzlich werden in diesem Kapitel häufig publizierte Konzepte untersucht: Administration, Delegation von Rollen bzw. Zugriffsrechten, negative Zugriffsrechte, dynamische Konzepte der Subjekt-, Zugriffsrechtzuordnung und Domänenbeschränkung sowie ausgewählte Strukturkonzepte von Rollen.

4.1 Rollenkonzept versus Gruppenkonzept

Das Rollenkonzept des RBAC-Referenzmodells wird manchmal fälschlicherweise synonym mit dem Gruppenkonzept verwendet (Anderson 2001, S. 54; Barkley 1997, S. 129–130; Seufert 2001, S. 112–114). Im Folgenden wird eine Abgrenzung der beiden Konzepte vorgenommen (Seufert 2001, S. 112–114; Gebel 2003, S. 27) und der Einsatz in Betriebssystemen und Datenbankmanagementsystemen herausgearbeitet.

4.1.1 Abgrenzung von Rollen und Gruppen

Das Gruppenkonzept (siehe Kapitel 3.2.1.4) erweitert das Zugriffsmatrix-Modell. Die Gruppenzuordnung wird in der Regel von der Systemadministration vorgenommen, während die Zugriffsrechte für einzelne Dateien dezentral durch einzelne Subjekte vergeben werden. Eine Gruppe kann folgendermaßen charakterisiert werden:

- Eine Gruppe repräsentiert eine Menge bzw. eine Liste von Subjekten oder Gruppen, die gemeinsam administriert werden (Gebel 2003, S. 7; Jajodia et al. 2001, S. 220).
- Eine Gruppenmitgliedschaft besteht immer und jedes Gruppenmitglied hat dieselben Zugriffsrechte (Barkley 1997, S. 129–130).
- Alle Zugriffsrechte, die ein Subjekt über die Mitgliedschaft in einer Gruppe erlangt, stehen immer zur Verfügung (Barkley 1997, S. 130).

- Gruppen werden oftmals zur Vereinfachung der Administration gebildet (Seufert 2001, S. 114).
- Zugriffsrechte werden nicht nur durch die Administration vergeben, sondern auch direkt von Subjekten.
- Eine Gruppe ist normalerweise nicht leer und beinhaltet mindestens zwei Mitglieder (Sandhu 1996, S. I-25).
- Ein Subjekt kann direkt einer Gruppe zugeordnet sein oder indirekt über die Zuordnung einer Gruppe zu einer anderen Gruppe (Sandhu 1996, S. I-25).
- Das Zugriffsmatrix-Modell wird oftmals durch sog. Fähigkeitslisten implementiert. Dadurch ist es nur mit großem Aufwand möglich, herauszufinden, welche Gruppe ein bestimmtes Zugriffsrecht besitzt (Seufert 2001, S. 113). Damit ist keine einfache Protokollierung möglich.
- Zugriffsrechte können trotz der Existenz von Gruppen weiterhin direkt Subjekten zugeordnet werden (Seufert 2001, S. 113).

Rollen werden dagegen folgendermaßen beschrieben:

- Eine Rolle repräsentiert eine Menge von Zugriffsrechten (Jajodia et al. 2001, S. 220; Sandhu 1996, S. I-25).
- Rollen können aktiviert und deaktiviert werden (Jajodia et al. 1997, S. 33).
- Es stehen nur Zugriffsrechte der aktivierten Rollen zur Verfügung (Jajodia et al. 2001, S. 220).
- Rollen vereinfachen die Administration der Zugriffsrechte (Sandhu et al. 1996, S. 41).
- Zugriffsrechte werden fast ausschließlich über die Administration vergeben, außer es wird eine Delegation zugelassen.
- Einer Rolle muss nicht unbedingt ein Subjekt zugeordnet sein.
- Ein Subjekt kann direkt einer Rolle oder indirekt über die Rollenhierarchie einer Rolle zugeordnet sein.
- Die Protokollierung ist vom Modell her vorgegeben und kann damit leicht umgesetzt werden (Seufert 2001, S. 114).
- Zugriffsrechte können nie direkt einem Subjekt zugeordnet werden (Seufert 2001, S. 113).

Der Hauptunterschied zwischen Rollen und Gruppen ist, dass Gruppen als eine Zusammenfassung von Subjekten, während Rollen als eine Zusammenfassung von Zugriffsrechten angesehen werden. Eine Rolle dient als Zwischenglied zwischen einer Menge von Zugriffsrechten und einer Menge von Subjekten (Sandhu et al. 1996, S. 40). Ein weiterer Unterschied ist, dass Rollen durch ein Subjekt aktiviert und deaktiviert werden können, während eine Gruppenmitgliedschaft immer besteht. Die Menge der Zugriffsrechte hängt im Kontext der Rollen von der Aktivierung der potentiell möglichen Rollen ab (Anderson 2001, S. 54; Jajodia et al. 1997, S. 33–34). Dadurch wird das Prinzip der minimalen Rechte gewährleistet. Das Zugriffskontrollmodell RBAC erfüllt die Forderung der Informationssicherheit nach einer Protokollierung und sieht diese bereits im Modell vor und legt deren Implementierung fest.

Das Gruppenkonzept verfolgt eine dezentrale Administration, während RBAC eine zentrale verfolgt. Zwar werden die Gruppenzuordnungen hauptsächlich von der Administration vorgenommen, jedes Subjekt kann aber trotzdem seine Zugriffsrechte einzeln weitergeben. Das Referenzmodell von RBAC des ANSI-Standards sieht keine Möglichkeit vor, dass ein Subjekt Zugriffsrechte weitergibt. Allerdings kann dies bei Bedarf durch die Erweiterung um eine Delegation ermöglicht werden (siehe Kapitel 4.5).

4.1.2 Rollen- und Gruppenkonzept in Betriebs- und Datenbankmanagementsystemen

In diesem Abschnitt wird dargestellt, wie das Rollen- bzw. Gruppenkonzept in ausgewählten Betriebssystemen bzw. in Datenbankmanagementsystemen (DBMS) umgesetzt ist.

Die Betriebssysteme Windows 2000 und Unix unterstützen beide das Rollen- und das Gruppenkonzept. Windows 2000 unterstützt zwei Modus. Einmal den mit Windows NT4 kompatiblen Modus und den reinen Windows 2000 Modus. Im Windows NT4 Modus wird keine Hierarchie von Gruppen unterstützt. Der Windows 2000 Modus unterstützt hingegen eine Rollenhierarchie bzw. Zugriffsrechtsvererbung. In beiden Modus werden jedoch alle Gruppen eines Nutzers gleichzeitig in einer Sitzung aktiv. Selbst in Windows 7 wird nicht das Rollenkonzept angewendet,

sondern Zugriffsrechte weiterhin über Gruppenrichtlinien an Nutzer vergeben (o. V. 2014a).

Das Unix-Betriebssystem unterstützt ebenfalls das Gruppenkonzept ohne Möglichkeit der Hierarchiebildung und der Aktivierung einer einzelnen Gruppe.

Damit wird bei Betriebssystemen eher das Gruppenkonzept als das Rollenkonzept angewandt, auch wenn bei Windows der Begriff Rolle dafür verwendet wird. Außerdem wird das Konzept der Aufgabentrennung bei keinem der beiden Betriebssysteme umgesetzt (Essmayr et al. 2004, S. 136–137).

Bei der Analyse der beiden DBMS Informix und Oracle ergibt sich folgendes Bild: Bei Informix kann eine m:n-Beziehung zwischen Rollen und Subjekten abgebildet werden. Ein Subjekt kann zu einem Zeitpunkt nur eine Rolle aktivieren. Es besteht in Informix die Möglichkeit, Rollenhierarchien (nested roles) abzubilden, aber keine statische Aufgabentrennung. Die dynamische Aufgabentrennung ist indirekt implementiert, da es keine Aktivierung von mehr als einer Rolle gibt. Auch Oracle unterstützt eine m:n-Zuordnung von Subjekten zu Rollen. Es lässt die Aktivierung aller oder auch Teilmengen der zugeordneten Rollen zu. Eine Abbildung einer Rollenhierarchie ist möglich, eine Definition der Aufgabentrennung hingegen nicht (Essmayr et al. 2004, S. 137–139).

Zusammenfassend lässt sich sagen, dass in DBMS-Systemen das Rollenkonzept eher Anwendung findet als in Betriebssystemen. Das RBAC-Referenzmodell wird in Ansätzen verwendet, jedoch nicht komplett umgesetzt (Essmayr et al. 2004, S. 139).

4.2 Beschränkungen im RBAC-Modell

Im Folgenden wird eine ausführliche Betrachtung des Kontrollprinzips der Aufgabentrennung im Rahmen des RBAC-Modells vorgenommen sowie die Möglichkeit der zeitlichen Beschränkung vorgestellt.

4.2.1 Aufgabentrennung

Grundlage dieses Kapitels bilden die allgemeinen Beschreibungen der Aufgabentrennung in Kapitel 2.4.3 und im RBAC-Referenzmodell (siehe Kapitel 3.2.7.4). Das Konzept der Aufgabentrennung ist von Anfang an fester Bestandteil des Zugriffskontrollmodells RBAC (Ferraiolo et al. 1995, S. 246–247; Sandhu et al. 1996, S. 43–

44). Darauf aufbauend fasst dieser Abschnitt[34] ausgewählte weiterführende Ergebnisse zur Aufgabentrennung in RBAC zusammen (Giuri und Iglio 1997; Gligor et al. 1998; Ferraiolo et al. 2003, S. 91–112; Simon und Zurko 1997). (Simon und Zurko 1997) liefert einen umfassenden Überblick über die verschiedenen Typen von Aufgabentrennung[35]. Darauf aufbauend werden formale und praktische Aspekte der Problematik der Aufgabentrennung in realen Anwendungssystemen diskutiert (Ferraiolo et al. 2003, S. 110–112). Alle Veröffentlichungen haben folgende grobe Einteilung gemeinsam:

- Statische Aufgabentrennung als starker Ausschluss und
- Dynamische Aufgabentrennung als schwacher Ausschluss.

Die **Statische Aufgabentrennung** wird durch einen exklusiven Ausschluss zwischen zwei Rollen realisiert, indem eine Zuordnung beider Rollen zu ein und demselben Subjekt nicht erlaubt ist. Da es potentiell zu Inkonsistenzen zwischen einer Rollenhierarchie und einer statischen Aufgabentrennung kommen kann, muss bei der Implementierung eines Zugriffskontrollsystems darauf geachtet werden, dass eine statische Aufgabentrennung auch bei der Erstellung einer Rollenhierarchie (siehe Kapitel 4.3.2) überprüft wird (Ferraiolo et al. 2003, S. 102–103).

> *In einem Prüfungsverwaltungssystem ist eine statische Aufgabentrennung sowohl zwischen Studierenden und Prüfungsamt als auch zwischen Lehrstühlen und Prüfungsamt notwendig. Ein Subjekt ist entweder im Prüfungsamt beschäftigt oder ein Studierender. Dies ist der funktionale Aspekt einer Aufgabentrennung. Lehrstühle sind verantwortlich für die Eingabe der Noten, während das Prüfungsamt die Noten bestätigt. Erst nach der Bestätigung der Noten durch das Prüfungsamt sind die Noten für den Studierenden sichtbar. Dies bezieht sich auf das 4-Augenprinzip.*

Es existieren aber organisatorische Fälle, in denen die statische Aufgabentrennung zu starr ist, um die Bedürfnisse einer Organisation abzubilden. Deshalb wird auf die dynamische Aufgabentrennung zurückgegriffen. Eine **dynamische Aufgabentren-**

[34] Eine formale Darstellung der Aufgabentrennung basierend auf Simon und Zurko (1997) einschließlich des Kontextes des Anwendungssystems findet sich in Gligor et al. (1998). Die Aufgabentrennung wurde als anwendungsspezifische Sicherheitsrichtlinie auch im Zusammenhang mit Transaktionen unabhängig von RBAC diskutiert Clark und Wilson (1987); Nash und Poland (1990), Sandhu (1988, 1991), Thomas und Sandhu (1994).

[35] Zusätzliche Aspekte finden sich in Ferraiolo et al. (2003, S. 91–113); Giuri und Iglio (1997); Gligor et al. (1998).

nung ist eine flexiblere Möglichkeit, um die Aktivierung und den Gebrauch von Rollen zu kontrollieren. Sie erlaubt der Administration, Subjekten Rollen, die sich ausschließen, zuzuordnen solange verhindert wird, dass diese in der derselben Sitzung aktiviert werden können. Innerhalb der dynamischen Aufgabentrennung existieren folgende sechs Aspekte (Simon und Zurko 1997):

- **Einfache dynamische Aufgabentrennung**: Bei der Subjektzuordnung werden Subjekte den entsprechenden Rollen zugeordnet. Es wird dafür gesorgt, dass sich ausschließende Rollen nicht zur selben Zeit aktiviert werden können (Ferraiolo et al. 2003, S. 98; Simon und Zurko 1997, S. 186).
- **Objektbasierte Aufgabentrennung**: Rollen können gemeinsame Subjekte haben und diese Rollen auch gleichzeitig aktivieren, aber kein Subjekt kann dasselbe Objekt mehrmals bearbeiten (Ferraiolo et al. 2003, S. 100–101; Nash und Poland 1990, S. 203; Simon und Zurko 1997, S. 186).
- **Operationale Aufgabentrennung**: Unter operationaler Aufgabentrennung wird die Umsetzung des 4-Augenprinzips verstanden. Es ist einem Subjekt allein nicht erlaubt, alle Aktionen, die erforderlich sind, um eine kritische Aufgabe zu beenden, auszuführen. Implizit wird damit festgelegt, dass mindestens zwei Rollen notwendig sind, um eine kritische Aufgabe auszuführen. Eine operationale Aufgabentrennung kann jedoch nicht nur dynamisch, sondern auch statisch erfolgen (Ferraiolo et al. 2003, S. 99–100; Simon und Zurko 1997, S. 186). Zur Umsetzung des 4-Augenprinzips ist eine Betrachtung der Historie der erfolgten Zugriffe erforderlich (Bussler 1998, S. 345).
- **Vergangenheitsorientierte Aufgabentrennung**: Bei einer objektbasierten sowie operationalen Aufgabentrennung wird die Autorisierung eines Zugriffs auf ein Objekt in Abhängigkeit von einem Zugriff in der Vergangenheit auf dieses Objekt bestimmt. Deshalb können diese beiden als vergangenheitsorientierte Aufgabentrennung zusammengefasst werden. Diese Art der Aufgabentrennung erhöht die Komplexität, weil ein Zugriffskontrollsystem erforderlich ist, dass alle Aktivitäten der Subjekte auf Objekten nicht nur protokolliert, sondern auch auswertet (Sandhu 1988, S. 282; Simon und Zurko 1997, S. 186; Ferraiolo et al. 2003, S. 100–101).
- **Reihenfolgeunabhängige Aufgabentrennung:** Im Referenzmodell wird eine reihenfolgeunabhängige Aufgabentrennung angenommen. Es werden die Zu-

griffsrechte zur Sicherstellung der Aufgabentrennung auf mehrere Rollen verteilt, aber die Reihenfolge in denen Subjekte die jeweiligen Aktionen ausführen, ist dabei nicht von Bedeutung.

- **Reihenfolgeabhängige Aufgabentrennung:** Es existieren durchaus Situationen, in denen eine bestimmte Reihenfolge bei der Ausführung von Rollen eingehalten werden muss. Wenn die Aufgabentrennung allein durch Verteilen der Zugriffsrechte auf mehrere Rollen nicht ausreicht, um Missbrauch zu verhindern, muss die Reihenfolgebeziehung bei der Aufgabentrennung berücksichtigt werden (Simon und Zurko 1997, S. 186).

Die Aufgabentrennung subsumiert verschiedene Aspekte und kann auf unterschiedlichste Weise implementiert werden. Mit der Verwendung des RBAC-Referenzmodells kann bereits im Modell sowohl statische als auch dynamische Aufgabentrennung berücksichtigt werden. Vergangenheitsbezogene bzw. reihenfolgeabhängige Aspekte der Aufgabentrennung müssen, wenn erforderlich, durch die Implementierung umgesetzt und bei der Protokollierung und Rechteprüfung berücksichtigt werden.

4.2.2 Zeitliche Beschränkungen

Neben statischer und dynamischer Aufgabentrennung können in bestimmten Anwendungsszenarien zeitabhängige Beschränkungen notwendig sein (Bertino et al. 2001; Ferraiolo et al. 2003, S. 112–113; Joshi et al. 2005)[36]. RBAC lässt sich mit zeitlichen Beschränkungen um den Zeitaspekt erweitern. Zeitabhängige Beschränkungen lassen sich in drei Kategorien einteilen:

- Zeitliche Beschränkungen der Rolle
- Zeitliche Beschränkungen der Subjektzuordnung
- Zeitliche Beschränkungen der Zugriffsrechtzuordnung (Ferraiolo et al. 2003, S. 113–115).

Eine Zeitangabe kann hierbei durch Festlegung einer Periode oder einer Dauer vorgenommen werden. Die Beschränkungen können sich auf die Freigabe einer Rolle oder auf die Aktivierung bzw. Deaktivierung von Rollen beziehen (Ferraiolo et al. 2003, S. 114–116).

[36] Für eine ausführliche Behandlung wird auf Bertino et al. (2001); Joshi et al. (2005); Ferraiolo et al. (2003, S. 112–116) verwiesen.

4.3 Konzept der Rollenhierarchie im RBAC-Modell

In diesem Kapitel werden Rollenhierarchien unter verschiedenen Aspekten betrachtet. Grundlage für weiterführende Überlegungen bildet die Definitionen (siehe Kapitel 3.2.7.3) für Rollenhierarchien des Referenzmodells. Diese wird um virtuelle Rollen und regionale Aspekte erweitert. Anschließend werden die intensionalen und extensionalen Aspekte einer Rollenhierarchie analysiert. Nach der Beschreibung verschiedener Arten von Rollenhierarchien werden anschließend Rollenhierarchie und Aufgabentrennung in Beziehung gesetzt und dahingehend bewertet, wie eine Kombination von Rollenhierarchie und Aufgabentrennung am besten realisiert werden kann.

4.3.1 Allgemeine Überlegungen zu Rollenhierarchien

Neben der Abbildung der Hierarchie aus dem Geschäftsprozess und der Organisation existieren überlappende Verantwortlichkeiten und allgemeine Berechtigungen wie z. B. E-Mail abrufen oder Sitzung aktivieren, die unabhängig von der zu erledigenden Aufgabe sind (Ferraiolo et al. 1995, S. 244). Um zu verhindern, dass jeder Rolle immer wieder eine Menge allgemeiner Zugriffsrechte zugeordnet werden müssen, werden spezielle Rollen in die Rollenhierarchie eingebunden, die gemeinsame Zugriffsrechte auf einer Ebene kapseln. Diese Rollen werden **virtuelle Rollen** genannt (Ferraiolo et al. 2003, S. 76; Moffett 1998, S. 65). Diese virtuellen Rollen werden keinem Subjekt zugeordnet. Als Faustregel gilt: falls eine mindestens 80%ige Übereinstimmung der Zugriffsrechte zwischen Rollen besteht, ist dies ein Indiz für die Einführung einer virtuellen Rolle (Ferraiolo et al. 2003, S. 79).

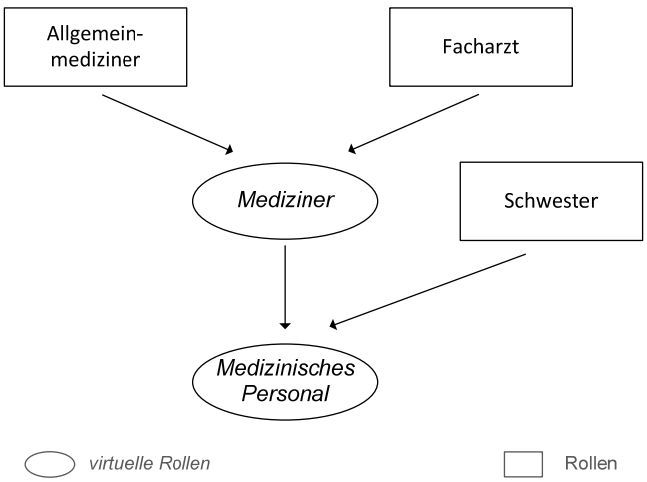

Abb. 4-1 Beispiel einer Rollenhierarchie (Ferraiolo et al. 1995, S. 244)

*Die Rolle **Medizinisches Personal** kapselt z. B. alle grundlegenden Zugriffsrechte eines Subjektes in einem Krankenhaus. Abb. 4-1 zeigt eine solche Hierarchie: dabei sind **Medizinisches Personal** und **Mediziner** virtuelle Rollen.*

Weder bestimmt alleine die Organisationsstruktur die Rollenhierarchie, noch beschreibt die Organisationsstruktur die Zugriffsrechtsvererbungsbeziehung. Die Betrachtung von Aufgaben, Geschäftsprozessen sowie beteiligten Aufgabenträgern helfen Rollenhierarchien zu bilden. In weltweit operierenden Unternehmen beinhalten Rollenhierarchien zudem noch regionale Aspekte (Ferraiolo et al. 2003, S. 80). Um regionale Aspekte zu berücksichtigen, kann das Konzept der parametrisierten Rolle durch Attribute (siehe Kapitel 4.7) genutzt werden.

Beispielsweise kommt die Rolle Buchhalter in verschiedenen regionalen Niederlassungen vor. Um diese zu vereinheitlichen, gibt es unternehmensweit eine Rolle Buchhalter, diese hat jedoch Attribute wie z. B. Ort, Organisationseinheit, Filiale und Region, um dadurch einen regionalen Bezug herzustellen (Ferraiolo et al. 2003, S. 81–82).

Beim Einsatz des rollenbasierten Zugriffskontrollmodells in Workflow Management Systemen sollte nicht die Aufbauorganisation allein durch die Rollenhierarchie abgebildet werden, sondern auch die Aufgabenverantwortlichkeiten innerhalb des Geschäftsprozesses.

Der Buchhalter ist zwar organisatorisch dem Controller unterstellt, aber der Controller sollte nicht zur Elternrolle des Buchhalters gemacht werden (Ferraiolo et al. 2003, S. 216–217), da hierdurch die Aufgabentrennung verletzt wird.

4.3.2 Intensionaler und extensionaler Aspekt der Rollenhierarchie

Nachfolgend wird das Konzept der Intention und Extension auf Rollenhierarchien übertragen. Im Kontext der objektrelationalen Datenbanken werden u. a. der extensionale als auch der intensionale Aspekt beim Konzept der Subklassenbildung (Spezialisierung) beschrieben (Türker und Saake 2006, S. 111-113; 126-128). „Die intensionale Spezialisierung erfolgt auf der Ebene der Datentypen“ (Türker und Saake 2006, S. 112). Im Falle der Rollenhierarchie bilden die Zugriffsrechte die

Merkmale, die betrachtet werden. Die Zugriffsrechte sind die Eigenschaften einer Rolle und stellen den intensionalen Aspekt der Rollenhierarchie dar.

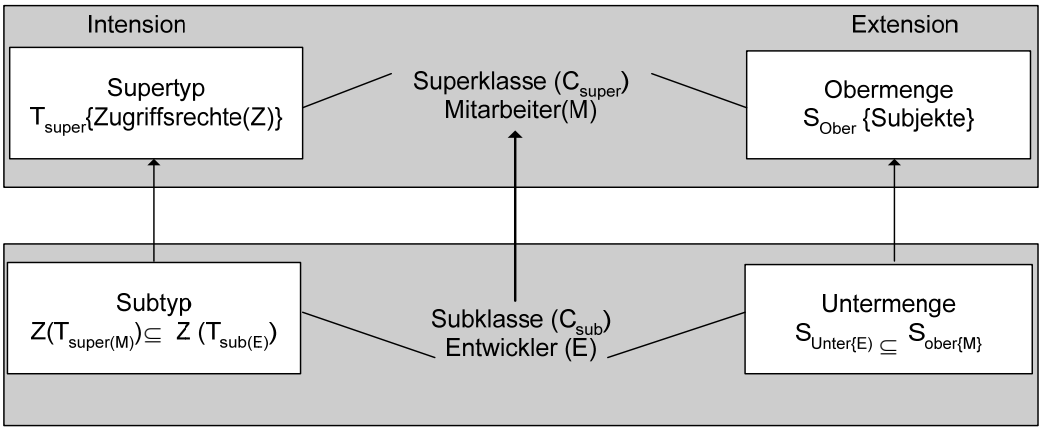

Abb. 4-2 Intension und Extension der Rollenhierarchien nach (Türker und Saake 2006, S. 68)

„Die extensionale Spezialisierung geschieht auf Basis der Datenmengen“ (Türker und Saake 2006, S. 112). Übertragen auf das Konzept der Rollenhierarchie bedeutet dies, der Blickwinkel der Subjektmitgliedschaft ist der extensionale Aspekt der Rollenhierarchie. Subjekte können als Instanziierung der Rolle, also als Objekte von Rollen betrachtet werden Der intensionale Aspekt bildet die Zugriffsrechtsvererbung ab, der extensionale Aspekt betrachtet die Vererbung der Subjektmitgliedschaft.

*Als Beispiel zeigt **Abb. 4-2** die beiden Rollen **Mitarbeiter** und **Entwickler**. Die Rolle **Mitarbeiter** ist die allgemeine Rolle, die Zugriffsrechte kapselt, die jeder Mitarbeiter benötigt. Der extensionale Aspekt, die Vererbung der Subjektmitgliedschaft, beschreibt, dass alle Subjekte, die Mitglieder der Rolle **Entwickler** sind, auch Mitglieder der Rolle **Mitarbeiter** sind. Die Spezialrolle **Entwickler** hat weniger Mitglieder als die allgemeine Rolle **Mitarbeiter**. Der intensionale Aspekt, die Zugriffsrechtsvererbung sagt aus, dass die Rolle **Entwickler** alle Rechte der Rolle „Mitarbeiter“ besitzt. Die Rolle **Entwickler** hat darüber hinaus noch weitere Zugriffsrechte. Die Menge der Zugriffsrechte des Supertyps **Mitarbeiters** sind eine Teilmenge der Zugriffsrechte des Subtyps **Entwickler**.*

4.3.3 Ausgewählte Rollenhierarchien

In diesem Kapitel werden ausgewählte Konzepte zu Rollenhierarchien vorgestellt, um anschließend das Thema Aufgabentrennung und Rollenhierarchien zueinander in Beziehung zu setzen. Folgende drei Typen von Rollenhierarchien werden untersucht:

- is_a – Rollenhierarchie, basierend auf Generalisierung

- Aktivitäten Rollenhierarchie, basierend auf Aggregation
- Aktivierungshierarchie (Sandhu 1998)

is_a-Rollenhierarchie

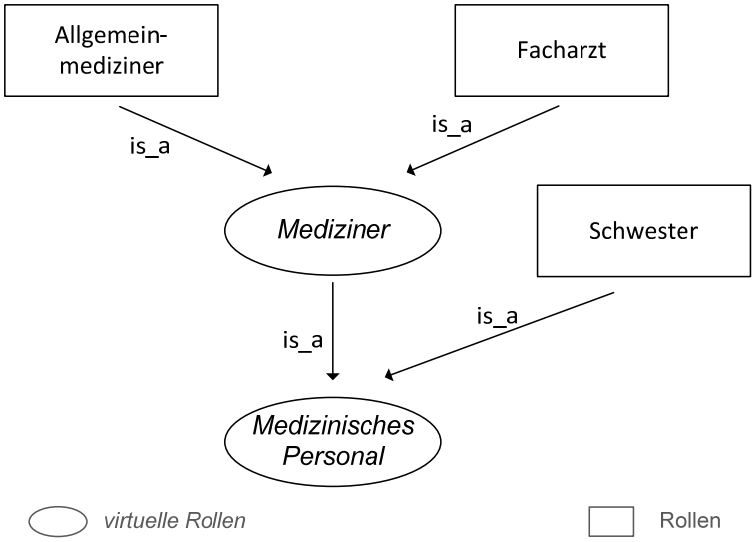

Abb. 4-3 is_a – Rollenhierarchie nach (Moffett 1998, S. 65)

Abb. 4-3 zeigt eine is_a-Rollenhierarchie aufgrund von Kompetenz und Aufgaben. Ein Allgemeinmediziner **is_a** Mediziner **is_a** *Medizinisches Personal*. Rollen können in diesem Konzept auch virtuelle Rollen sein, die dieselben Zugriffsrechte kapseln (Moffett 1998, S. 65).

Aktivitäten Rollenhierarchie

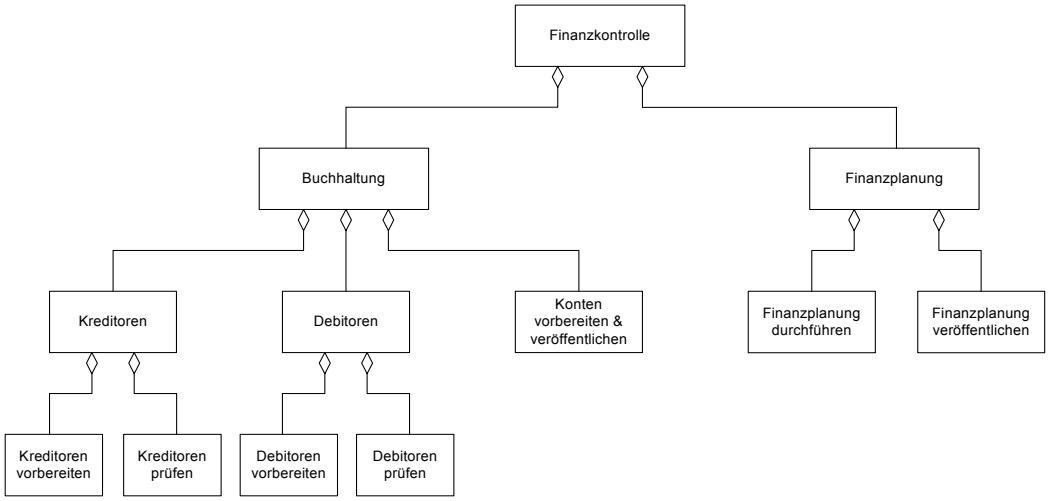

Abb. 4-4 Aktivitäten Rollenhierarchie nach (Moffett und Lupu 1999, S. 155)

Eine Aktivitäten Rollenhierarchie entsteht aus der is part of – Beziehung. Diese Form der Hierarchie basiert auf Aktivitäten und den Beziehungen innerhalb der Organisation: „IstVerantwortlichFür“ bzw. „FührtDurch“. Ist ein Subjekt für eine Aktivität

verantwortlich, dann erledigt es diese Aufgabe selbst oder delegiert diese. Auf diese Art und Weise entsteht eine Zerlegung innerhalb der Rollenhierarchie. Beispiel einer Aktivitäten Rollenhierarchie zeigt **Abb. 4-4**. Je höher eine Rolle in der Rollenhierarchie angesiedelt ist, desto mehr Aktivitäten und Verantwortung sind in dieser Rolle vereinigt. Eine Aktivitäten-Rollenhierarchie muss nicht zwingend der Aufbauorganisation eines Unternehmens entsprechen (Moffett 1998, S. 65).

Aktivierungshierarchie

Um eine dynamische Aufgabentrennung in Rollenhierarchien sicherzustellen, gibt es den Ansatz, zwischen den Aspekten der Zugriffsrechtsvererbung und Aktivierung von Rollen in einer Hierarchie zu unterscheiden. Die Mitgliedschaft in einer Elternrolle autorisiert normalerweise dessen Rolleninhaber, auch die Kindrollen zu aktivieren. Um das Prinzip der minimalen Zugriffsrechte sowie die Aufgabentrennung zu gewährleisten, sollte es möglich sein, nur eine bestimmte Rolle z. B. die Kindrolle zu aktivieren und die Elternrolle als schlafend zu kennzeichnen (Sandhu 1998, S. 33). Parallel zur Rollenhierarchie, die die Zugriffsrechtsvererbung abbildet, wird eine weitere Hierarchie implementiert, die festlegt, welche Rollen in der Hierarchie gemeinsam aktiviert werden dürfen. Liegt keine dynamische Aufgabentrennung zwischen Rollen vor, so können alle Rollen von der Elternrolle aus aktiviert werden. Liegt eine dynamische Aufgabentrennung vor, so können Rollen nicht gemeinsam aktiviert werden und es muss dann explizit möglich sein, nur eine Kindrolle zu aktivieren oder zu deaktivieren (Sandhu 1998, S. 38).

4.3.4 Problematik von Rollenhierarchie und Aufgabentrennung

Das wichtige Kontrollprinzip Aufgabentrennung ist ohne Definition einer Hierarchie eindeutig zu überprüfen. Aber auch bei Verwendung von Rollenhierarchien muss die Aufgabentrennung berücksichtigt werden. Ein möglicher Konflikt entsteht, wenn über einer Hierarchie einem Aufgabenträger zwei konkurrierende Rollen zugeordnet werden. Damit kann es zwischen der Aufgabentrennung und der Modellierung einer Rollenhierarchie in RBAC zu unerwünschten Nebeneffekten kommen.

Bei der Modellierung der Rollenhierarchie bzw. bei der Subjektzuordnung muss die statische Aufgabentrennung berücksichtigt werden, indem der Rollenadministration z. B. ein Warnhinweis gegeben wird, falls ein Subjekt von zwei Rollen im Pfad der Hierarchie erbt, die sich aber gegenseitig ausschließen. Innerhalb einer Rollenhierar-

chie mit sich ausschließenden Rollen darf es keinen „Superuser“ geben, der alle Rollen beinhaltet, da die Aufgabentrennung schon bei Bildung der Hierarchie berücksichtigt werden muss (Ferraiolo et al. 2003, S. 102–103). Schwieriger ist die Überprüfung bei der dynamischen Aufgabentrennung, da hier die Überprüfung erst mit der Aktivierung der Rollen erfolgt.

Mögliche Lösungen zum Vermeiden solcher Konflikte sind:

- Keine Definition von Rollenhierarchien zulassen, was aber den Aufwand für die Administration erhöht.
- Keine Rollen vererben über die eine Aufgabentrennung vorgenommen worden ist. Daraus ergibt sich das Problem, dass zuerst die Aufgabentrennung feststehen muss, um anschließend eine Hierarchie modellieren zu können. Spätere Änderungen der Aufgabentrennung können zu Fehlern führen.
- Einführung von privaten Rollen, in denen alle Rechte gekapselt werden, die nicht vererbt werden dürfen.
- Es wird nur die is_a-Hierarchie verwendet, die sich aus virtuellen Rollen zusammensetzt. Diese Hierarchie dient als ein Beschreibungsverfahren zum Kapseln von Zugriffsrechten, um den Aufwand der Subjektzuordnungen gering zu halten.
- Es wird eine strenge Vererbung nach dem Prinzip der Delegation vorgenommen und somit eine Aktivitäten-Rollenhierarchie erzeugt.
- Es werden zwei Rollenhierarchien modelliert: eine Zugriffsrechtsvererbungshierarchie und getrennt davon eine Aktivierungshierarchie.

Für die Implementierung und Überwachung einer dynamischen Aufgabentrennung bietet sich die Bildung einer is_a-Hierarchie einschließlich virtueller Rollen als Kapselung von Zugriffsrechten an. Diese wird ausschließlich als Zugriffsrechtsvererbungshierarchie interpretiert und implementiert. Eine zweite davon getrennte Rollenhierarchie dient als Aktivierungshierarchie, die die Aufgabentrennung bei der Aktivierung der Rollen berücksichtigt.

4.4 Administration im rollenbasierten Zugriffskontrollmodell

Die Administration von Zugriffskontrollsystemen kann als sicherheitskritisch eingestuft werden und muss sorgfältig kontrolliert werden, damit die Ziele der Sicherheits-

strategie eingehalten werden (Sandhu et al. 1999, S. 106). Deshalb wird vorgeschlagen die Administration im RBAC-Referenzmodell zentral von einer Sicherheitsadministration durchzuführen.

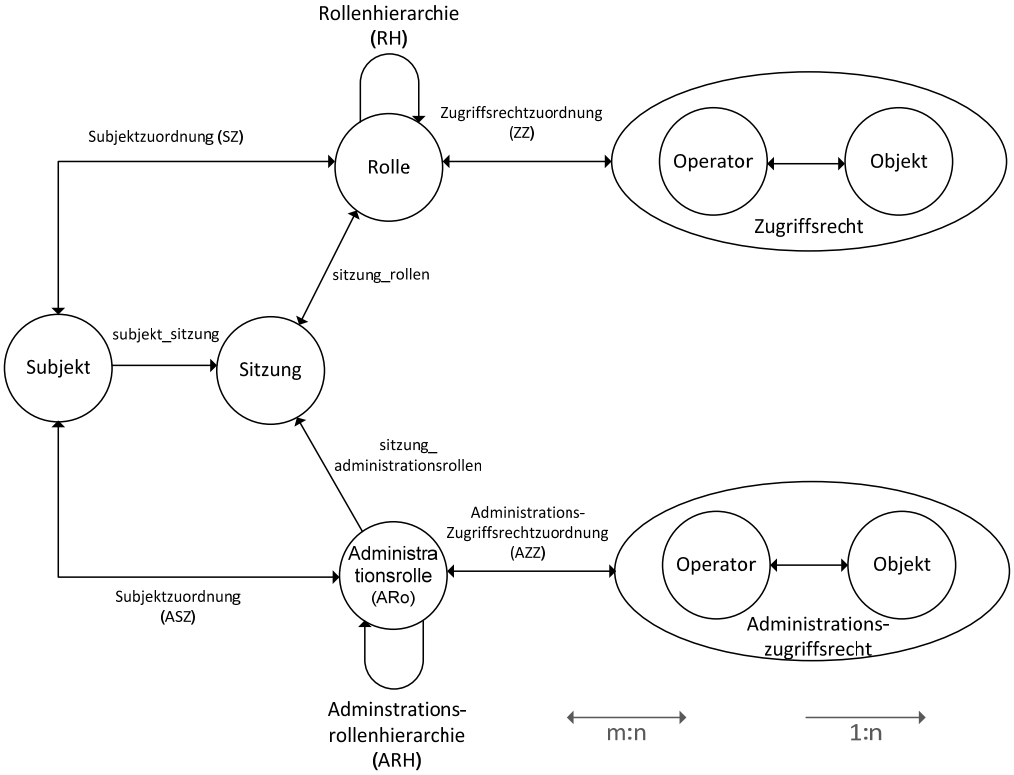

Abb. 4-5 Administration als Erweiterung des RBAC nach (Sandhu et al. 1999, S. 108)

Einer der Vorteile des RBAC-Modells ist die Vereinfachung der Administration (Sandhu et al. 1996, S. 81). Deshalb ist es naheliegend das Konzept der Rollen ebenfalls zu nutzen, um die Rechteverwaltung von Zugriffskontrollsystemen zu untergliedern und auf mehrere Administratoren zu verteilen (Sandhu et al. 1999, S. 106). Das u.a. entwickelte ARBAC97, das die Grundlagen einer Administration mit RBAC beschreibt (Sandhu et al. 1999) wird im Folgenden vorgestellt.

Neben den Rollen und Zugriffsrechten für das zu überwachende Anwendungssystem werden eine Menge von Administrationszugriffsrechten und Administrationsrollen, die eine Administrationsrollenhierarchie (ARH) bilden können, definiert, die disjunkt von der Menge der Anwendungsrollen sind. Subjekte und Sitzung hingegen werden gemeinsam sowohl für die Administration als auch für die zu kontrollierenden Anwendungssysteme verwendet (Sandhu et al. 1999, S. 107). Dieser Zusammenhang wird in **Abb. 4-5** visualisiert.

4.4.1 Grundlagen für die Administration

Die Definitionen der einzelnen Komponenten des RBAC-Referenzmodells wurden für die Administration angepasst. Die Administrationszugriffsrechte beziehen sich auf Objekte und den dazugehörigen Operatoren für die Administration. Administrationsrollen (ARo) besitzen über eine Administrationszugriffsrechtszuordnung (AZZ) nur Administrationszugriffsrechte (AZ). Die Subjektzuordnung ordnet den Subjekten Administrationsrollen zu. Die Administrationsrollenhierarchie wird analog der Rollenhierarchie im Referenzmodell gebildet und bezieht sich ausschließlich auf Administrationsrollen (Sandhu et al. 1999, S. 108).

Es existieren zwei Basiskomponenten, um Administrationsregeln (A_{Regeln}) definieren zu können. Diese werden auch im Kapitel 4.5 der Delegation wieder aufgegriffen:

- Eine Vorbedingung (VB) legt fest, für welche Subjekte bzw. Zugriffsrechte eine Operation ausgeführt werden darf.
- Rollenbereiche (ROB) definieren den Geltungsbereich der Administrationsregeln anhand einer in der Bereichsnotation festgelegten Rollenmenge.

Definition 4-1 Vorbedingung nach (Sandhu et al. 1999, S. 110)

Eine Vorbedingung ist ein Boolescher Ausdruck unter der Verwendung des logischen UND-Operators $\wedge$ oder des logischen ODER-Operators $\vee$. Die Operatoren werden auf den Termen der Form ro und $\overline{ro}$ angewendet, wobei ro $\in$ Ro. Die Vorbedingung wird ausgewertet im Bereich der Subjektzuordnung (SZ) in Bezug auf ein bestimmtes Subjekt s $\in$ S für die Zuweisung eines Subjektes zu einer Rolle oder bei der Zugriffsrechtszuordnung (ZZ) auf ein bestimmtes Zugriffsrecht z $\in$ Z bei der Zuweisung eines Zugriffsrechts zu einer Rolle.

- Die Vorbedingung wird für ein Subjekt s bezüglich des Terms ro als wahr interpretiert, wenn gilt: $\exists$ (ro' $\geq$ ro) (s, ro') $\in$ SZ.
- Die Vorbedingung wird für ein Subjekt s bezüglich des Terms $\overline{ro}$ als wahr interpretiert, wenn gilt: $\exists$ (ro' $\geq$ ro) (s, ro') $\notin$ SZ.
- Die Vorbedingung wird für ein Zugriffsrecht z bezüglich des Terms ro als wahr interpretiert, wenn gilt: $\exists$ (ro' $\geq$ ro) (z, ro') $\in$ ZZ.
- Die Vorbedingung wird für ein Zugriffsrecht z bezüglich des Terms $\overline{ro}$ als wahr interpretiert, wenn gilt: $\exists$ (ro' $\geq$ ro) (z, ro') $\notin$ ZZ.

Definition 4-2 Rollenbereich nach (Sandhu et al. 1999, S. 113)

Ein Rollenbereich beschreibt eine Menge von Rollen mit der Bereichsnotation einer oberen ro_o und unteren ro_u Grenze. Es gilt:

– $[ro_u, ro_o] = \{ ro \in Ro \mid ro_u \leq ro \wedge ro \leq ro_o\}$ $[ro_u, ro_o) = \{ ro \in Ro \mid ro_u \leq ro \wedge ro < ro_o\}$

– $(ro_u, ro_o] = \{ ro \in Ro \mid ro_u < ro \wedge ro \leq ro_o\}$ $(ro_u, ro_o) = \{ ro \in Ro \mid ro_u < ro \wedge ro < ro_o\}$

Alle möglichen Rollenbereiche bei einer gegebenen Rollenhierarchie werden durch die Menge der Rollenbereiche ROB beschrieben: $ROB \rightarrow 2^{Ro}$.

Zusätzlich werden die Mitglieder einer Rolle danach unterschieden, ob ein Subjekt direkt oder über die Rollenhierarchie einer Rolle zugeordnet ist:

- Explizite Mitglieder werden die direkt einer Rolle zugeordneten Subjekte genannt.
- Implizite Mitglieder erben eine Rolle über die Rollenhierarchie (Sandhu et al. 1999, S. 116).

Es werden fünf Administrationsregeln eingeführt. Innerhalb einer Subjektzuordnung (ASZ) ist die Zuordnung von Subjekt zu Rollen ($vergabe_{Ro}$) und der Entzug einer Rollenmitgliedschaft ($entzug_{Ro}$) definiert. Innerhalb der Zugriffsrechtszuordnung ist Vergabe ($vergabe_Z$) bzw. Entzug ($entzug_Z$) des Zugriffsrechts definiert. Um Rollen hinzuzufügen, zu ändern und Veränderungen in einem Bereich der Rollenhierarchie vornehmen zu können, wird die Funktion ($änderung_{ROH}$) definiert (Sandhu et al. 1999, S. 110–113).

Um eine Verletzung der Sicherheitsstrategie bei der Administration zu vermeiden, wird eine Aufgabentrennung festgelegt. Es werden dazu getrennte Administrationsbereiche eingeführt:

- Die Zugriffsrechtszuordnung (AZZ) ist eine sicherheitssensitive Operation und wird innerhalb des Administrationsschemas separat betrachtet.
- Die Rollenverwaltung dient dem Löschen und Anlegen von Rollen und deren Einbindung in die Rollenhierarchie (Sandhu et al. 1999, S. 109–133).
- Die Subjektzuordnung (ASZ) wird ebenfalls in einen eigenen Bereich ausgelagert.

4.4.2 Administrationsbereich der Zugriffsrechtszuordnung

Innerhalb der Zugriffsrechtszuordnung sind die erlaubten Operationen die Erstellung von Zugriffsrechten und die Bündelung von Zugriffsrechten zu Rollen ($vergabe_Z$) bzw. das Entfernen einer Zugriffsrechtszuordnung ($entzug_Z$). Die Vorbedingung zieht dazu die Zuordnung bzw. Nichtzuordnung eines Zugriffsrechts zu einer Rolle heran. Die Zuweisung und der Entzug von Zugriffsrechten werden hier ebenfalls getrennt (Sandhu et al. 1999, S. 120). Dafür gilt folgende Definition.

Definition 4-3 Erlaubnis einer Zugriffsrechtszuordnung nach (Sandhu et al. 1999, S. 110)

- Die Rollenzuordnung zu einem Zugriffsrecht $z \in Z$ wird kontrolliert mit Hilfe der Relation: $vergabe_Z \subseteq ARo \times VB \times ROB$.

  Zum Beispiel legt ein Tupel $vergabe_Z (aro_1, vb = ro_6 \vee ro_9, (ro_1, ro_3))$ fest, dass ein Subjekt, das die Administrationsrolle aro_1 explizit oder implizit besitzt, Zugriffsrechte an beliebige Rollen, die zwischen ro_1 und ro_3 liegen, zuordnen darf. Die Zuordnung ist auf jene Zugriffsrechte beschränkt, die bereits an die Rolle ro_6 oder die Rolle ro_9 gebunden sind.

Der Entzug eines Zugriffsrechts ist nicht an Vorbedingungen geknüpft, sondern hängt allein von dem expliziten oder impliziten Besitz der Administrationsrolle ab (Sandhu et al. 1999, S. 120). Für den Entzug eines Zugriffsrechts gilt folgende Definition:

Definition 4-4 Erlaubnis des Entzuges nach (Sandhu et al. 1999, S. 116)

- Der Entzug wird kontrolliert mit Hilfe der Relation: $entzug_Z \subseteq ARo \times ROB$.

  Zum Beispiel legt ein Tupel $entzug_Z (aro_1, (ro_1, ro_3))$ fest, dass ein Subjekt, das die Administrationsrolle aro_1 explizit oder implizit besitzt, die Rechte von beliebigen Rollen, die innerhalb des Bereiches (ro_1, ro_3) liegen, entziehen darf.

4.4.3 Administrationsbereich der Rollenverwaltung

Der Bereich der Rollenvergabe umfasst das Erzeugen und Löschen von Rollen und der Definition von Vererbungsbeziehungen (Sandhu et al. 1999, S. 122). Bei der Rollenverwaltung wird die Zuständigkeit für eine Teilhierarchie anhand von Rollenbereichen definiert. Durch das Hinzufügen bzw. Löschen von Rollen kann sich eine Veränderung der Grenzen eines Rollenbereichs ergeben. Die Regel $änderung_{ROH}$ definiert das Anlegen, Löschen und Einbinden von Rollen in die Rollenhierarchie in

den dafür definierten Bereichsgrenzen (Sandhu et al. 1999, S. 124). Die Rollenbereiche sind exklusiv der Bereichsgrenzen definiert. Im Gegensatz dazu werden die Rollenbereiche für Zugriffsrechtszuordnung bzw. Subjektzuordnung einschließlich der Bereichsgrenzen definiert.[37] (Sandhu et al. 1999, S. 131–133).

Die Bedeutung des Tupels (aro_{13}, (ro_{11}, ro_{31})) ist wie folgt: ein expliziter oder impliziter Besitzer der Administrationsrolle aro_{13}, kann Rollen in dem Bereich (ro_{11}, ro_{31}) anlegen und löschen, sowie die Beziehungen innerhalb dieses Bereiches verändern. Die Rollen ro_{11} bzw. ro_{31} gehören nicht zum Änderungsbereich.

4.4.4 Administrationsbereich der Subjektzuordnung

Innerhalb der Subjektzuordnung sind die erlaubten Operationen die $vergabe_{Ro}$ und $entzug_{Ro}$. Die Definitionen gelten analog der Zugriffsrechtszuordnung.

Zum Beispiel legt ein Tupel $vergabe_{Ro}$ (aro_1, $vb = ro_6 \wedge \overline{ro_4}$, ($ro_1$, ro_3)) fest, dass ein Subjekt, welches die Administrationsrolle aro_1 explizit oder implizit über die Rollenhierarchie besitzt, Subjekten, denen schon die Rolle ro_6 und nicht die Rolle ro_4 zugeordnet ist, eine beliebige Rolle, die innerhalb des Bereiches (ro_1, ro_3) liegt, zuordnen darf.

Bei der Subjektzuordnung wird nicht nur die Relation $vergabe_{Ro}$ berücksichtigt, sondern es müssen alle Bedingungen, wie z. B. Aufgabentrennung, die im Zugriffskontrollsystem für ein Anwendungssystem hinterlegt sind, erfüllt sein. Nur wenn die Relation $vergabe_{Ro}$ **und** auch alle Bedingungen erfüllt sind, kann eine Zuordnung erfolgen.

Der Entzug einer Rolle ist nicht an Vorbedingungen verknüpft, sondern hängt allein vom Besitz der Administrationsrolle ab.

Zum Beispiel legt ein Tupel $entzug_{Ro}$ (aro_1, (ro_1, ro_3)) fest, dass ein Subjekt, das die Administrationsrolle aro_1 direkt explizit oder implizit besitzt, eine beliebige Rolle, die innerhalb des Bereiches (ro_1, ro_3) liegt, entziehen darf.

4.4.5 Kritik und Erweiterung des Administrationsmodells

Durch die Aufteilung der Administration in die Bereiche Subjektzuordnung bzw. Zugriffsrechtszuordnung und durch die Definition von Vorbedingungen und Rollen-

[37] Für die ausführliche formale Herleitung wird auf Sandhu et al. (1999, S. 126–133) verwiesen.

bereichen wurde zwar ein Fortschritt für die Administration erreicht, es existieren dennoch einige Defizite (Oh und Sandhu 2002, S. 155; Kern et al. 2003, S. 10).
Schwachstellen in der Subjektzuordnung sind (Oh und Sandhu 2002, S. 158):

- Mehrstufige Subjektzuordnung: Sind an der Zuordnung durch die Einteilung der Rollenbereiche mehrere Administratoren beteiligt, muss die Subjektzuordnung mehrstufig nacheinander erfolgen, um einem neuen Subjekt eine Rolle in der Hierarchie zuordnen zu können.
- Redundante Subjektzuordnungen: Zu einer Redundanz von Subjektzuordnungen kann es bei Subjekten kommen, die Rollen administrieren. Administratoren erhalten Rollen nicht um Zugriffsrechte auszuführen, sondern um damit Rollenbereiche definieren zu können, damit die Administration Subjektzuordnungen vornehmen kann.
- Eingeschränkte Zuordnungsmöglichkeiten von Subjekten: Sowohl die Rollen zur Definition der Vorbedingung als auch die Rollen der Subjektzuordnung beziehen sich auf dieselbe Rollenhierarchie. Dadurch kommt es zu eingeschränkten Kompositionsmöglichkeiten von Subjekten und Rollenhierarchien und kann eine Reorganisation erschweren (Oh und Sandhu 2002, S. 158).

Schwachstellen in der Zugriffsrechtszuordnung finden sich analog der Subjektzuordnung (Oh und Sandhu 2002, S. 158):

- Mehrstufige Zugriffsrechtszuordnung
- Redundante Zugriffsrechtszuordnung
- Eingeschränkte Komposition der Zugriffsrechtszuordnung
- Unbeschränkte Zugriffsrechtszuordnung
- Unerwünschte Seiteneffekte

Die Probleme der mehrstufigen und redundanten Zugriffsrechtszuordnung sowie die eingeschränkte Komposition der Zugriffsrechtszuordnung sind analog zu den Schwachstellen bei der Subjektzuordnung. Außerdem ist es nicht möglich, einer neuen Rolle neue, nur für diese gültige Zugriffsrechte zuzuordnen. Es können nur Zugriffsrechte einer Rolle im Rollenbereich zugeordnet werden, deshalb existieren keine Zugriffsrechte nur für eine Rolle. Durch sich überschneidende Definitionen von Rollenbereichen kann es zu einer illegalen Zuordnung von Zugriffsrechten und somit zu unerwünschten Seiteneffekten kommen (Oh und Sandhu 2002, S. 158).

Die Ursache dieser Schwachstellen ist, dass die zur Verfügung stehenden Subjekte und Zugriffsrechte von den modellierten Rollen bzw. der Rollenhierarchie abhängen (Oh und Sandhu 2002, S. 158). Ein Vorschlag, um die genannten Schwachstellen zu überwinden, ist die Weiterentwicklung des ARBAC97 über das ARBAC99-Modell (Sandhu und Munawer 1999) zum ARBAC02-Modell (Oh und Sandhu 2002). In ARBAC02 wurden dazu die folgenden Erweiterungen gewählt:

Es wird eine Rollenhierarchie basierend auf der Organisationsstruktur eines Unternehmens anstatt einer bestehenden Rollenhierarchie verwendet, um festlegen zu können, wer neue Subjekte anlegen und einordnen kann. Anstatt von Rollenbereichen für die Zuordnung von Subjekten bzw. Zugriffsrechten, die sich innerhalb der Rollenhierarchie befinden, wird ein davon getrennter Subjektpool bzw. Zugriffsrechtspool verwendet, um zu definieren, welche Subjekte bzw. Zugriffsrechte bearbeitet werden dürfen, um so das Problem der Mehrstufigkeit zu mindern (Oh und Sandhu 2002, S. 159).

4.5 Ausgewählte Delegationsmodelle für das RBAC-Modell

Die Grundidee einer Delegation ist die Übertragung von Zuständigkeiten, Weisungen und Befugnissen von einer Entität auf eine andere (Barka und Sandhu 2000b, S. 168), damit diese Aufgaben anstatt des Delegierenden ausführen kann. Eine Delegation in Informationssystemen kann zwischen folgenden verschiedenen Typen von Entitäten stattfinden:

- von Subjekt an Subjekt (Abadi et al. 1993, S. 2–3; Barka und Sandhu 2000b, S. 168),
- von Subjekt an Maschine (Abadi et al. 1993, S. 2–3; Barka und Sandhu 2000b, S. 168),
- von Maschine an Maschine (Abadi et al. 1993, S. 2–3; Barka und Sandhu 2000b, S. 168) und
- von Maschine an Subjekt (Barka und Sandhu 2000b, S. 168).

Auch wenn unterschiedliche Typen von Entitäten betroffen sind, entsteht immer dieselbe Wirkung – die Weitergabe von Zugriffsrechten (Zhang et al. 2003a, S. 405). Im Folgenden werden auf Grundlage von RBAC verschiedene Varianten der Delegation von Zugriffsrechten ausschließlich zwischen Subjekten untersucht.

Die Delegation[38] wurde nicht in das RBAC-Referenzmodell übernommen. Um eine Rolle im Referenzmodell an ein anderes Subjekt zu delegieren, muss der Delegierende die Administration beauftragen, dem Subjekt die gewünschte Rolle zuzuordnen. Dadurch entsteht ein erhöhter Aufwand aufgrund der einzuschaltenden Administration (Zhang et al. 2003a, S. 406).

Die in Kapitel 4.4 beschriebenen Grundlagen, wie Vorbedingungen bzw. explizite und implizite Mitgliedschaft von Rollen, werden auf den Bereich der Delegation angewendet. Nach einem Überblick über die Charakteristika einer Delegation werden verschiedene Delegationsmodelle vorgestellt.

4.5.1 Grundlagen der Delegation und Rücknahme von Rollen

Es existieren neben Implementierungsvorschlägen für auf RBAC basierende Delegationen u. a. ein Rahmenwerk für Delegation und Rücknahme (Barka und Sandhu 2000b) und eine Klassifikation der Rücknahme von Delegation (Hagström et al. 2001). Daraus können zusammenfassend folgende Merkmale für eine Delegation festgehalten werden:

- Dauer: Eine Delegation kann dauerhaft oder nur zeitlich beschränkt sein.
- Monotonie: Eine monotone Delegation bedeutet, dass der Delegierende die Zugriffsrechte der delegierten Rolle behält. Bei einer nicht-monotonen Delegation verliert der Delegierende die Möglichkeit, die delegierten Zugriffsrechte auszuüben.
- Gesamtheit: Gesamtheit bezieht sich darauf, ob alle oder nur Teile der Zugriffsrechte der delegierten Rolle delegiert werden.
- Administration: Die Delegation wird entweder subjektbestimmt vom Subjekt, dem eine Rolle zugeordnet ist oder administrationsbestimmt durch die Administration vorgenommen.
- Delegationsstufen: Eine Delegation kann einstufig oder mehrstufig durchgeführt werden. Mehrstufig bedeutet, dass der Delegierte Zugriffsrechte bzw. Rollen an weitere Subjekte delegieren kann.
- Mehrfache Delegation: Es kann eine Rolle vom Delegierenden an mehr als ein Subjekt delegiert werden oder ein Subjekt kann von mehr als einem Delegierenden für eine Rolle delegiert werden.

[38] Folgende Autoren beschäftigen sich mit den Aspekten der Delegation: Barka und Sandhu (2000a, 2000b, 2004), Hagström et al. (2001), Zhang et al. (2001, 2003a), Zhang et al. (2003b).

- Vereinbarung: Es kann zwischen unilateraler und bilateraler Delegation unterschieden werden. Bei einer bilateralen Delegation müssen beide Partner der Delegation zustimmen (Barka und Sandhu 2000b, S. 170–171).

Diese Merkmale können theoretisch beliebig kombiniert werden. Welche Kombinationen sinnvoll sind, wird ausführlich in (Barka und Sandhu 2000b) untersucht.[39]

Wird das RBAC-Referenzmodell um die Möglichkeit einer Delegation erweitert, muss auch die Gegenfunktion, die Rücknahme einer Delegation mit ihren verschiedenen Möglichkeiten untersucht werden, da eine Delegation normalerweise nicht unbegrenzt vergeben wird. Folgende Typen der Rücknahme sind möglich:

- Timeout: Zu einem festgelegten Zeitpunkt wird eine Delegation gelöscht.
- Gewährungsunabhängig: Bei einer gewährungsunabhängigen Rücknahme darf jedes Originalmitglied der delegierten Rolle die Delegation zurücknehmen.
- Gewährungsabhängig: Bei einer gewährungsabhängigen Rücknahme einer Delegation kann nur der Delegierende eine Delegation zurücknehmen (Barka und Sandhu 2000a, S. 5–6).

Vorteil einer Zurücknahme durch Timeout ist, dass dieser Prozess automatisch ausgeführt wird. Allerdings gewährleistet ein Timeout allein noch nicht die Informationssicherheit. Existiert außer dem Timeout kein weiterer Rücknahmemechanismus, so können, falls Delegierte sich nicht sicherheitskonform verhalten, die Sicherheitsrichtlinien verletzt werden. Außerdem muss der Zeitpunkt des Timeouts so gewählt werden, damit eine Delegation nicht zu lange oder zu kurz erfolgt.

Bei einer gewährungsunabhängigen Rücknahme muss keine Relation definiert werden, da alle Subjekte, die Originalmitglied einer delegierten Rolle sind, die Delegation zurücknehmen dürfen. Daraus können sich jedoch Konflikte unter den Originalmitgliedern ergeben (Barka und Sandhu 2000a, S. 7).

Bei einer gewährungsabhängigen Zurücknahme einer Delegation werden Konflikte zwischen den Originalmitgliedern zu Lasten einer höheren Komplexität vermieden:

- Es muss gespeichert werden, wer die Delegation vorgenommen hat.

[39] Für ein vertieftes Studium wird auf Barka und Sandhu (2000b, S. 173–175) verwiesen.

- Es muss das weitere Vorgehen definiert werden, wenn der Delegierende die delegierte Rolle verliert.
- Das Verfahren bei einer mehrfachen Delegation an einen Delegierten muss definiert werden (Barka und Sandhu 2000a, S. 6–7).

Bei einer Zurücknahme einer Delegation muss auch die Weiterreichung dieser Zurücknahme einbezogen werden. Für eine explizit gelöschte Rollenzuordnung gibt es drei Weitergabemöglichkeiten:

- Wird dem Delegierenden die delegierte Rolle entzogen, werden allen Subjekten, die dieser delegierten Rolle vom Delegierenden zugeordnet wurden, die entsprechenden Rollen entzogen.
- Entzieht der Delegierende die Rolle dem Delegierten, verlieren auch alle Subjekte diese delegierte Rolle, an die der Delegierte diese weitergegeben hat (mehrstufige Zurücknahme).
- Delegierte verlieren eine delegierte Rolle, falls einem Delegierten die Rolle entzogen wird, die dazu geführt hat, dass die Vorbedingung erfüllt war, um die delegierte Rolle zu erhalten (Barka und Sandhu 2000a, S. 7).

Orthogonal zu den beschriebenen Vorgehensmöglichkeiten kann eine schwache bzw. starke Rücknahme sowohl in Rollenhierarchien als auch bei mehrfacher Delegation unterschieden werden (Hagström et al. 2001, S. 49–54):[40]

- Starke Rücknahme bedeutet, die Rücknahme einer expliziten Rolle wird über die gesamte Rollenhierarchie bzw. über alle Delegationsstufen hinweg weitergereicht.
- Bei einer schwachen Zurücknahme wird nur die explizit delegierte Rolle entfernt (Hagström et al. 2001, S. 49–54).

Nach diesen allgemeinen Betrachtungen werden im Folgenden drei Delegationsmodelle untersucht:

- Rollenbasiertes Delegationsmodell - RBDM0 und RBDM1 (Barka und Sandhu 2000a; Barka und Sandhu 2000b; Barka und Sandhu 2004),
- Rollenbasiertes Delegationsmodell - RDM2000 (Zhang et al. 2001; Zhang et al. 2003a),

[40] Für eine ausführliche Beschreibung der Klassifikation der Rücknahme einer Delegation wird auf Hagström et al. (2001) verwiesen.

- Zugriffsrechtbasiertes Delegationsmodell - PBDM (Zhang et al. 2003b)[41].

4.5.2 Rollenbasierte Delegationsmodelle - RBDM0 und RBDM1

Das Kernmodell des RBAC-Referenzmodells wird zum rollenbasierten Delegationsmodell RBDM0 und RBDM1 erweitert. Das RBDM1-Modell betrachtet zusätzlich noch die Rollenhierarchie (Barka und Sandhu 2004).

RBDM0

Bei einer Delegation wird die Subjektzuordnung vom jeweiligen Subjekt ohne Einbindung einer Administration durchgeführt. Um das Modell einfach zu halten, wurden für das RBDM0-Modell folgende Annahmen getroffen:

- Grundlage ist das RBAC-Modell von (Sandhu et al. 1996) ohne Rollenhierarchie, welches etwa dem Kernmodell des Referenzmodells entspricht (Barka und Sandhu 2000a, S. 3).
- Delegation zwischen Subjekten, die dieselbe Rolle besitzen, ist nicht erlaubt.
- Es wird nur eine einstufige Delegation zugelassen.
- Rollen können nur mit allen Zugriffsrechten delegiert werden.
- Im Grundmodell wird nur eine gewährungsunabhängige Zurücknahme betrachtet.
- Jede Delegation hat ein Zeitelement, eine sog. Dauer (T) (Barka und Sandhu 2000a, S. 4).

Der Prozess der Delegation wird mit der Funktion $\text{delegation}_{Ro} \subseteq Ro \times Ro$, analog der Funktion vergabe_{Ro} im ARBAC (siehe Kapitel 4.4) kontrolliert.

> *Die Bedeutung des Tupels $(ro_1, ro_3) \in \text{delegation}_{Ro}$ sagt aus, dass ein Subjekt, das ein Originalmitglied der Rolle ro_1 ist, diese an ein beliebiges anderes Subjekt, das Originalmitglied von ro_3 ist, delegieren darf (Barka und Sandhu 2000a, S. 5).*

Zusätzlich wird als eine mögliche Erweiterung die gewährungsabhängige Zurücknahme betrachtet (Barka und Sandhu 2000a, S. 6–7). Außerdem wird vorgeschlagen, die Zugriffsrechte in delegierbare und nicht delegierbare Zugriffsrechte zu untertei-

[41] Für formale Definitionen und ausführliche Beschreibungen wird auf die jeweilige Literatur verwiesen.

len, damit die Administration Einflussmöglichkeit erhält, welche Zugriffsrechte einer Rolle delegiert werden können (Barka und Sandhu 2000a, S. 8).

RBDM1

Das RBDM1-Modell fügt dem RBDM0-Modell die Rollenhierarchie hinzu und legt folgende zusätzliche Annahme fest. Delegation erfolgt in der Hierarchie nach unten oder quer zur Hierarchie, da eine Elternrolle bereits alle Zugriffsrechte aus den Kindrollen besitzt (Barka und Sandhu 2004, S. 399).

Neben den Originalmitgliedern SZ_O und den delegierten Mitgliedern SZ_D wird eine weitere Unterteilung in explizite und implizite Subjekte vorgenommen (siehe Kapitel 4.4). Es verändert sich durch Einführung der Rollenhierarchie die Semantik der Relation $delegation_{Ro}$. In Abhängigkeit von einer expliziten bzw. impliziten Originalmitgliedschaft des Delegierenden bzw. des Delegierten muss die Relation = $(ro_i, ro_{ii}) \in delegation_{Ro}$ unterschiedlich interpretiert werden[42].

> *Das Tupel $(ro_1, ro_3) \in delegation_{Ro}$ sagt aus, dass ein Subjekt, das ein explizites oder implizites Originalmitglied der Rolle ro_1 ist, diese an ein beliebiges anderes Subjekt, das ein explizites oder implizites Originalmitglied von ro_3 ist, eine explizite delegierte Mitgliedschaft an einer Rolle $ro' \geq ro_1$ erzeugen darf (Barka und Sandhu 2004, S. 401).*

Die Zurücknahme der Delegation durch Subjekte wird im RBDM1 unter folgenden Aspekten und Annahmen betrachtet:

- Eine Zurücknahme durch Timeouts erfolgt analog zum RBDM0-Modell.
- Es wird nur eine gewährungsabhängige Rücknahme der Delegation erlaubt.
- Es wird eine stufenweise Zurücknahme unterstützt.
- Die Rücknahme der Delegation muss entsprechend für eine mehrfache Delegation gestaltet sein. Im RBDM1-Modell wird nur die starke Rücknahme betrachtet (Barka und Sandhu 2004, S. 402–403).

4.5.3 Rollenbasiertes Delegationsmodell RDM2000

Das RDM2000-Modell unterstützt hierarchische Rollen und mehrstufige Delegation und erweitert RBDM0 um Regeln, die sowohl eine Delegation als auch eine Rück-

[42] Für die ausführliche Beschreibung wird auf Barka und Sandhu (2004, S. 401) verwiesen.

nahme der Delegation durchführen und beschränken (Zhang et al. 2001)[43]. Folgende Annahmen an eine Delegation werden definiert:

- Es können nur reguläre Rollen, jedoch keine Administrationsrollen delegiert werden.
- Jede Subjektzuordnung ist eindeutig. Hat ein Subjekt schon explizit oder implizit die Rolle ro_1, so kann die Rolle ro_1 nicht an das Subjekt delegiert werden.
- Das Modell erlaubt eine mehrstufige Delegation durch Definition einer maximalen Tiefe, die bei einer mehrstufigen Delegation erreicht werden darf (Zhang et al. 2001, S. 155).
- Es wird eine administrationsbestimmte Delegation vorgeschlagen, d. h. die Administration definiert in welchen Bereichen eine Delegation möglich ist (Zhang et al. 2003a, S. 426).
- Die Delegation kann auch nur einen Teil der Zugriffsrechte einer Rolle betreffen.
- Die Aufgabentrennung aus dem RBAC-Modell muss auch bei einer Delegation eingehalten werden (Zhang et al. 2003a, S. 408).

Die Bedeutung des Tupels $(ro_1, vb, n) \in delegation_{Ro}$ sagt aus: Ein Subjekt, das ein explizites oder implizites Mitglied der Rolle ro_1 ist, kann die Rolle ro_1 oder eine Kindrolle von ro_1 an ein beliebiges Subjekt delegieren, dessen gegenwärtige Rollenberechtigungen die Vorbedingung vb erfüllen und die maximale Tiefe des Delegationspfades nicht überschreitet (Zhang et al. 2003a, S. 413).

Tab. 4-1 Festlegungen für die Rücknahme einer Delegation in RDM2000

Subjekt der Rücknahme	Gewährungsabhängig	Gewährungsunabhängig
Auswirkung auf die Rollenhierarchie	Starke Rücknahme	Schwache Rücknahme
Tiefe der Rücknahme	Einstufige	Mehrstufige

Aus den in **Tab. 4-1** vorgestellten Möglichkeiten ergeben sich insgesamt acht verschiedene Kombinationen, eine Delegation wieder zurückzunehmen. Zusätzlich

[43] Für die formale Beschreibung wird auf Zhang et al. (2003a) verwiesen.

zu den gewählten Verfahren benötigt eine Rücknahme einer Delegation eine Autorisierung (Zhang et al. 2003a, S. 414; 416).[44]

> *Die Bedeutung von $ro_3 \in rücknahme_{GDRo}$ ist, dass nur das delegierende Subjekt, das gegenwärtig Mitglied der Rolle ro_3 ist, einem Delegierten eine Rolle, die eine Vorgängerrolle von ro_3 ist, entziehen kann. Die Bedeutung von $ro_3 \in rücknahme_{GIRo}$ ist, dass ein beliebiges Subjekt, das eine gegenwärtige Mitgliedschaft einer delegierten Rolle ro_3 im Delegationspfad einschließt, die delegierte Rolle ro_3 dem Delegierten entziehen kann (Zhang et al. 2003a, S. 416).*

Die Umsetzung organisatorischer und administrativer Regeln, die überprüfen, ob es einem Subjekt erlaubt ist, eine Delegation oder eine Rücknahme vorzunehmen, erfolgt im RDM2000 mittels einer deklarativen regelbasierten Sprache (Zhang et al. 2003a, S. 418).[45]

4.5.4 Zugriffsrechtsbasiertes Delegationsmodell PBDM

Die beiden bisher vorgestellten Delegationsmodelle erlauben eine Delegation ganzer Rollen mit allen Zugriffsrechten. Um auch einzelne Zugriffsrechte zu delegieren, wurden die Modelle RBDM0 und RDM2000 zu einem zugriffsrechtsbasierenden Delegationsmodell (Permission-based Delegation Modell: PBDM) erweitert (Zhang et al. 2003b). Zusammenfassend lassen sich die Modelle PBDM0 bis PBDM1 wie folgt beschreiben:

- Es wird einstufige und mehrstufige Delegation unterstützt.
- Es ist möglich, sowohl eine Delegation von Rollen, als auch von einzelnen Zugriffsrechten vorzunehmen (Zhang et al. 2003b, S. 150).

Um einzelne Zugriffsrechte delegieren zu können, benötigt der Delegierende im Unterschied zu den bisher beschriebenen Modellen, die Möglichkeit, Rollen anzulegen.

PBDM0

In diesem Modell werden Rollen unterteilt in reguläre Rollen (Ro_R) und Delegationsrollen (Ro_D). Aufgrund dieser Aufteilung folgt eine Trennung der Subjektzuordnung

[44] Für ein tiefer gehendes Studium und Beispiele wird auf die Literatur Zhang et al. (2003a, S. 414–417) verwiesen.

[45] Für ein weiterführendes Studium des Aufbaus der regelbasierten Sprache wird auf Zhang et al. (2003a, S. 418–425) verwiesen.

(SZ_R, SZ_D) und der Zugriffsrechtszuordnung (ZZ_R, ZZ_D). Eine reguläre Rolle wird als eine dauerhafte Rolle und eine Delegationsrolle als temporäre Rolle betrachtet. Eine Delegationsrolle kann nicht in eine reguläre Rollenhierarchie eingeordnet werden. Damit wird verhindert, dass delegierte Zugriffsrechte durch die Rollenhierarchie vererbt werden. Die Definition regulärer Rollen, die Subjektzuordnung sowie die Einordnung in die Rollenhierarchie liegen ausschließlich im Aufgabenbereich der Administration, während eine entsprechende Definition der Delegationsrollen und der dazugehörigen Zuordnung im Verantwortungsbereich des delegierenden Subjekts liegt (Zhang et al. 2003b, S. 150–151).

Jede Delegationsrolle hat einen Eigentümer und nur dieser darf eine Delegationsrolle löschen. Eine Rollenhierarchie von Delegationsrollen ist nur im Delegationsbereich eines Eigentümers möglich. Für die Überprüfung, ob eine Delegation vorgenommen werden darf, wird eine $delegation_{Ro}$ definiert, die sich an den vorherigen Konzepten orientiert.

Eine Zurücknahme der Delegation beinhaltet drei Fälle:

- Die Subjektzuordnung (SZ_D) zwischen einem Subjekt und einer Delegationsrolle wird gelöscht.
- Die Zugriffsrechtszuordnung (ZZ_D) zwischen der Delegationsrolle und dem Zugriffsrecht wird geändert, um ein oder mehrere Zugriffsrechte zu entziehen.
- Die Delegationsrolle wird gelöscht (Zhang et al. 2003b, S. 150).

PBDM1

In PBDM1 kann zusätzlich von der Administration festlegen werden, welche Rollen bzw. Zugriffsrechte delegiert werden dürfen. Die Rollen werden zusätzlich noch unterteilt in reguläre Rollen, delegierbare Rollen und Delegationsrollen. Dabei sind reguläre und delegierbare Rollen als dauerhafte Rollen anzusehen. Jede delegierbare Rolle basiert auf einer regulären Rolle, wobei eine 1:1 Beziehung zwischen diesen besteht. Das Einfügen von delegierbaren Rollen sowie Subjektzuordnungen und Zugriffsrechtszuordnung kann nur von der Administration vorgenommen werden. Damit wird sichergestellt, dass die Administration die Zugriffsrechte kontrolliert, die in eine Delegationsrolle einfließen können. Alle Subjekte müssen sowohl zur regulären

Rolle als auch zu der korrespondierenden delegierbaren Rolle zugeordnet werden. (Zhang et al. 2003b, S. 152).

Zusätzlich zu den Verfahren der Zurücknahme einer Delegation durch das Subjekt besteht beim PBDM1-Modell[46] die Möglichkeit der Einflussnahme der Administration. Sie hat folgende Rücknahmemöglichkeiten:

- Eine Subjektzuordnung (SZ_R, SZ_{DB}) zwischen Subjekten und delegierbaren Rollen wird gelöscht.
- Eine Zugriffsrechtszuordnung (ZZ_{DB}) zwischen delegierbaren Rollen und Zugriffsrechten wird geändert, um ein oder mehrere Zugriffsrechte zu entfernen (Zhang et al. 2003b, S. 153).

4.5.5 Zusammenfassung

Die ausgewählten Delegationsmodelle erweitern das Referenzmodell von RBAC um Delegation. Die Modelle reichen von Delegation von ganzen Rollen bis hin zur Delegation einzelner Zugriffsrechte mit und ohne Ausbildung einer Rollenhierarchie. Es wurden Delegationsmodelle vorgestellt, die Delegation nicht alleine dem Delegierenden zu überlassen, sondern die Sicherheitsadministration gibt den Rahmen der Delegationsmöglichkeiten vor. Aufbauend auf den theoretischen Grundlagen wurden bereits Delegationsklienten entwickelt, die für die Einrichtung, Überwachung und Durchführung einer administrationsbestimmten Delegation zuständig sind (Liebrand et al. 2002, S. 2).

Breiten Raum in den untersuchten Publikationen nimmt die Zurücknahme einer Delegation ein. Überlegungen dazu reichen von:

- automatischer Zurücknahme durch einen vorher eingestellten Endzeitpunkt,
- nur der Delegierende darf die Delegation zurücknehmen,
- alle Subjekte, denen die delegierte Rolle zugeordnet ist oder
- eine autorisierte Administration darf eine Delegation zurücknehmen.

Delegation ist eine Erweiterung des RBAC-Modells, um eine höhere Flexibilität bei der Zuordnung von Zugriffsrechten bzw. Rollen zu Subjekten zu erreichen. Damit die Sicherheitsrichtlinien eines Unternehmens oder einer Verwaltung nicht verletzt werden, sollte eine administrationsbestimmte Delegation gewählt werden.

[46] Für die formale Beschreibung des PDBM1-Modells sowie ausführliche Informationen wird auf Zhang et al. (2003b) verwiesen.

4.6 Negative Zugriffsrechte im RBAC-Modell

Die Konstruktionsprinzipien sicherer Zugriffskontrollsysteme (siehe Kapitel 2.4.3) empfehlen eine Autorisierung nach dem Erlaubnisprinzip innerhalb eines geschlossenen Systems. Im Gegensatz dazu wird in offenen Systemen der Zugriff nur verweigert, wenn ein explizites negatives Recht existiert (Jajodia et al. 2001, S. 228). Negative Zugriffsrechte erhöhen nicht die Mächtigkeit des Zugriffskontrollsystems, denn jede Strategie kann allein mit positiven Zugriffsrechten formuliert werden (Stiemerling et al. 2000, S. 321).

Es existieren jedoch geschlossene Zugriffskontrollsysteme, die sowohl positive als auch negative Zugriffsrechte erlauben. Wenn sowohl ein negatives als auch positives Zugriffsrecht auf dasselbe Objekt spezifiziert ist, können dabei Konflikte[47] entstehen, die durch entsprechende Regeln aufzulösen sind (Jajodia et al. 1997, S. 31). Dies wird durch Untersuchungen von negativen Zugriffsrechten bei relationalen Datenbankmanagementsystemen und benutzerbestimmter Zugriffskontrollstrategie bestätigt. Eine mögliche Konfliktlösungsstrategie ist, durch Hinzufügen eines negativen Rechtes ein vorhandenes gleiches positives Zugriffsrecht zu blockieren (Bertino et al. 1993, S. 130–131; Bertino et al. 1997b, S. 86). Wird ein negatives Zugriffsrecht, für das ein positives Zugriffsrecht existiert, entzogen, dann wird die Blockade des Zugriffsrechts aufgehoben und das Subjekt kann dieses Zugriffsrecht wieder ausüben. Damit kann auch ein zeitweiliger Entzug eines Zugriffsrechts erfolgen, ohne dass das positive Zugriffsrecht entfernt werden muss (Bertino et al. 1997b, S. 86).[48]

Das Referenzmodell des rollenbasierten Zugriffskontrollmodells kennt nur eine Autorisierung nach dem Erlaubnisprinzip. Ein Nachteil des Erlaubnisprinzips ist, dass das Fehlen eines Zugriffsrechts für ein gegebenes Subjekt das Subjekt nicht daran hindert, das Zugriffsrecht zu einem späteren Zeitpunkt fälschlicher Weise zu erlangen (Bertino et al. 1997b, S. 88). Beispielsweise darf das Subjekt **x** keinen Zugriff auf Objekt **o** erhalten. Bei dezentraler Zugriffsrechtsadministration oder Delegation kann dem Subjekt **x** das Zugriffsrecht für den Zugriff auf Objekt **o** durch einen Fehler gegeben werden. Aus diesem Grund wird in der Literatur vorgeschlagen, ein ex-

[47] Eine ausführliche Beschreibung von Konfliktlösungsstrategien findet sich in Moschgath (2002, S. 108–122).

[48] Für Beweise und formale Beschreibungen wird auf Bertino et al. (1993, 1997b) verwiesen.

plizites negatives Zugriffsrecht einzuführen, das sicherstellt, dass ein Subjekt niemals Zugriff auf ein bestimmtes Objekt erhält (Bertino et al. 1997b, S. 88).

Die allgemeinen Überlegungen zu negativen Zugriffsrechten wurden auf das rollenbasierte Zugriffskontrollmodell übertragen (Al-Kahtani und Sandhu 2004; Hagström et al. 2001). Mögliche Einsatzgebiete von negativen Rechten innerhalb des RBAC-Modells können im Bereich einer dezentralen Zuordnung von Zugriffsrechten (siehe Kapitel 4.4) zu Rollen, einer Delegation (siehe Kapitel 4.5) und einer dynamischen Subjektzuordnung als regelbasierten Zuordnung (siehe Kapitel 4.7) anhand von Attributen des Subjektes sein. Durch diese Erweiterungen kann es zu einer Verletzung der Zugriffskontrollstrategie kommen. Deshalb kann es von Vorteil sein, negative Rechte einzuführen, um die unternehmensweite Sicherheitsstrategie sicherzustellen.

Die Verwendung von negativen Zugriffsrechten im Zusammenhang mit Delegation kann z. B. folgende zwei Bedeutungen haben: Aufbauend auf den vorherigen Überlegungen können negative Zugriffsrechte dazu dienen, ein einmal delegiertes Zugriffsrecht inaktiv zu setzen und zurückzunehmen. Durch die Einführung von negativen Zugriffsrechten bleibt die Dokumentation einmal vergebener Zugriffsrechte für Rollen erhalten (Hagström et al. 2001, S. 45). Im zweiten Anwendungsfall drückt ein negatives Recht aus, dass eine bestimmte Rolle dieses Zugriffsrecht im Normalfall nicht ausführen darf. (Na und Cheon 2000).

Im Bereich der dynamischen Zuordnung von Subjekten zu Rollen können negative Zugriffsrechte sicherstellen, dass trotz einer dynamischen Zuordnung bestimmte Subjekte ein Zugriffsrecht nicht ausüben können, obwohl das Subjekt alle Attributwerte besitzt, die zu einer automatischen Subjektzuordnung führen. Dies wird erreicht, indem diesem Subjekt von der Administration das dazugehörige negative Zugriffsrecht zugeordnet wird (Al-Kahtani und Sandhu 2004, S. 409–410).

4.7 Dynamische Konzepte im RBAC-Modell

Weitere Konzepte für Erweiterungen des RBAC-Referenzmodells betreffen die Möglichkeit zur Laufzeit die Zugriffsrechtszuordnung und die Ausgabe von Informationen durch Domänenbeschränkungen dynamisch vorzunehmen (Kern und Walhorn 2005; Al-Kahtani 2003; Al-Kahtani und Sandhu 2004; Chandramouli 2000; Gebel 2003, S. 18–20; Ge und Osborn 2004; Giuri und Iglio 1997). Sowohl Attribute

des agierenden Subjektes als auch Attribute des Kontextes des ausführenden Prozesses sollten als Bestandteil eines Zugriffskontrollkonzeptes betrachtet werden (Kern et al. 2004, S. 87–88). Es werden nachfolgend drei ausgewählte Konzepte beschrieben:

- Kontextabhängiges RBAC,
- Dynamische attributabhängige und regelbasierte Subjektzuordnung und
- Domänenbeschränkung anhand von Parametern.

4.7.1 Kontextabhängiges RBAC

Im RBAC-Referenzmodell wird die Entscheidung des Zugriffs zustandsunabhängig vorgenommen. Um dieses statische RBAC-Referenzmodell zu einem dynamischen Zugriffskontrollmodell zu erweitern, werden kontextabhängige Informationen herangezogen. Kontextabhängige Informationen sind z. B. Zeit, Ort, Prozessstatus oder die Historie der Zugriffe. Mit Hilfe kontextabhängiger Informationen können komplexe Zugriffskontrollstrategien umgesetzt werden (Strembeck und Neumann 2004, S. 393; Beresnevichiene 2003, S. 47–48). Dies bedeutet, dass obwohl ein Subjekt ein Zugriffsrecht besitzt, der Zugriff aufgrund des speziellen Kontextes verweigert werden kann. Mögliche Beschränkungen im RBAC können Folgende sein:

- Endogene Beschränkungen, die im Modell bereits vorkommen, wie statische und dynamische Aufgabentrennung (siehe Kapitel 4.2.1).
- Exogene Beschränkungen umfassen Attribute, die nicht im RBAC-Modell vorkommen. Diese schränken den Zugriff zusätzlich ein und sind bei der Subjektzuordnung noch nicht bekannt (Strembeck und Neumann 2004, S. 395–396).

Diese Aufteilung ist nicht komplett orthogonal, zeigt aber die verschiedenen Aspekte auf (Strembeck und Neumann 2004, S. 396). Diese kontextabhängigen Beschränkungen können durch sog. Kontextbedingungen umgesetzt werden (Stiemerling et al. 2000, S. 399).

Ein Beispiel einer solchen exogenen Beschränkung ist, dass jeder Mitarbeiter einen Reisekostenvorschuss beantragen kann und der Vorgesetzte diesen genehmigen muss. Beantragt der Vorgesetzte selbst einen Reisekostenvorschuss, so darf er diesen nicht selbst genehmigen.

Um dies abbilden zu können, muss die Historie der Rollennutzung berücksichtigt werden. Dieser erweiterte Ansatz der dynamischen Rollenbeschränkung wird in der Literatur häufig als kontextabhängige Aufgabentrennung beschrieben. Erst durch die Berücksichtigung der Historie kann der dynamische Ansatz umgesetzt werden. Dieser flexible Ansatz führt zu einer größeren Komplexität in der Administration und Autorisierung des Zugriffsrechts, da die Autorisierung nicht mehr nur vom Zugriffskontrollmodell abhängt.

Eine weitere Möglichkeit für eine kontextabhängige Autorisierung ist die Einführung von Zugriffsregeln, die entscheiden, ob ein vom Modell gewährtes Zugriffsrecht auch unter dem Aspekt des Kontextes noch besteht. Mögliche Kontextbeschränkungen sind:

- zeitliche Beschränkungen,
- Interessenkonflikte aufgrund früherer Zugriffe,
- Aufgabentrennung aufgrund früherer Zugriffe,
- Vertrauensstellung des Subjektes innerhalb der Organisation und
- Parameter aus der Geschäftslogik (Chandramouli 2000, S. 10f).

Diese Beschränkungen können durch Zugriffsrechte mit Übergabeparametern abgebildet werden. Vor einem Zugriff wird anhand einer Regel, die dem Zugriffsrecht zugeordnet ist, entschieden, ob trotz Rollenzugehörigkeit der Zugriff, auch in Abhängigkeit vom Geschäftsprozess, erlaubt ist oder nicht.

4.7.2 Attributabhängige regelbasierte RBAC

Die Zugriffskontrolle in Anwendungssystemen sollte eine rollenbasierte Zugriffskontrolle unterstützen, jedoch ist ein Zugriffsrecht in Anwendungssystemen nicht immer nur von der Rolle, sondern oftmals auch von Attributen des Subjektes abhängig, auf denen Regeln angewendet werden. Dies bedeutet, dass eine Zugriffskontrolle Referenzen auf Werte von Subjekten und Objekten benötigt, die zum Zeitpunkt der Rollendefinition noch nicht bekannt sind (Kern et al. 2004, S. 90). Für diese Fälle wurden attributabhängige regelbasierte Konzepte und Implementierungen diskutiert. Es werden dabei der dynamische Aspekt der Subjektzuordnung und das dynamische Ausüben von Zugriffsrechten betrachtet.

Attributabhängige regelbasierte Subjektzuordnung

Unter dem Begriff attributabhängige regelbasierte Subjektzuordnung wird eine dynamische Subjektzuordnung, die sich an den Attributen des Subjektes orientiert und über Regeln vorgenommen wird, subsumiert (Al-Kahtani und Sandhu 2002; Al-Kahtani 2003; Al-Kahtani und Sandhu 2004; Kern und Walhorn 2005). Diese flexible Subjektzuordnung soll die Administration bei häufig wechselnden Subjektzuordnungen entlasten. Dafür werden den Subjekten Attribute zugeordnet. Unternehmensweit festgelegte Sicherheitsrichtlinien definieren die Regeln auf diesen Attributen und mit Hilfe einer regelbasierten Sprache werden diese ausgewertet und eine dynamische Subjektzuordnung vorgenommen (Al-Kahtani und Sandhu 2002, S. 255). Durch eine zusätzliche Erweiterung um negative Zugriffsrechte kann durch die Administration verhindert werden, dass ein Subjekt trotz Zuordnung über Regeln einen unerlaubten Zugriff erhalten kann[49] (Al-Kahtani und Sandhu 2004, S. 408). Die Definition der Zugriffsregeln ist der kritische Bereich bei einer automatischen Subjektzuordnung. Zudem müssen sowohl die Auswirkungen auf eine durch regelbasierte Subjektzuordnung entstandene als auch auf die vorgegebene Rollenhierarchie genau betrachtet werden (Al-Kahtani 2003, S. 73–82).

Ein Nachteil der dynamischen Subjektzuordnung zur Laufzeit ist, dass eine Protokollierung schwieriger durchzuführen ist. Es ist bspw. nicht leicht herauszufinden, ob einem Subjekt zum Zeitpunkt **t** eine bestimmte Rolle zugeordnet war (Kern und Walhorn 2005, S. 132). Diesen Nachteil will die folgende Variante vermeiden. Die Ermittlung der Subjektzuordnung läuft automatisch einmal täglich. Diese ermittelten Subjektzuordnungen werden gespeichert und die Zugriffsrechtszuordnung zu den Subjekten wird zur Laufzeit anhand der gespeicherten Subjektzuordnungen, wie beim RBAC-Referenzmodell, durchgeführt (Kern und Walhorn 2005, S. 132–133). Eine zuverlässige Protokollierung kann dabei nur gewährleistet werden, wenn eine Historisierung der durch den automatischen Lauf erzeugten Subjektzuordnungen stattfindet.

Dynamische Aktivierung der Zugriffsrechte über Attribute

Eine dynamische Aktivierung der Zugriffsrechte kann sich auf Attribute der folgenden Konstrukte im RBAC-Modell beziehen:

[49] Eine ausführliche Beschreibung findet sich in Al-Kahtani (2003).

- Subjekt: Die Attribute beschreiben das Subjekt näher und beschränken die Zugriffsrechte.
- Subjektzuordnung: Die Attribute sind nicht abhängig von einen Subjekt, sondern von der Zuordnung zur jeweiligen Rolle.
- Rolle: Die Attribute beziehen sich auf die Rolle.

Existiert zudem eine Rollenhierarchie, so muss geklärt werden, wie diese Attribute nun innerhalb einer Rollenhierarchie behandelt werden sollen:

- Die Subjektattribute sind gültig für alle Rollen, mit denen das Subjekt verbunden ist.
- Die Attribute werden nicht direkt einem Subjekt zugeordnet, sondern einer Subjektzuordnung. Dies gilt auch innerhalb der Rollenhierarchie. Werden Attribute bei Mehrfachvererbung über verschiedene Pfade unterschiedlich gesetzt, muss dafür ein Lösungsweg definiert werden. Ein möglicher Lösungsweg kann sein, diese Attribute zu akkumulieren.
- Attribute werden durch eine Rollenhierarchie nicht beeinflusst (Kern et al. 2004, S. 95).

Durch diese attributabhängige Subjektzuordnung und Aktivierung von Zugriffsrechten wird eine Dynamisierung des RBAC-Modells erreicht und die Administration entlastet. Dafür ist aber eine Definition der Sicherheitsrichtlinien in einer Regelsprache notwendig, damit es zu keiner Verletzung der Sicherheitsrichtlinien kommt.

4.7.3 Domänenbeschränkung im RBAC

Eine andere Art der Dynamisierung ist eine Domänenbeschränkung im RBAC. Diese ist dann sinnvoll, wenn Zugriffsrechte sich auf immer dieselben Objekte mit denselben Operatoren beziehen, aber die zu lieferden Domänen sich unterscheiden. Rollen reichen für die Ermittlung des Zugriffsrechts aus, aber sagen noch nichts über die Domäne aus, die bearbeitet wird.

*Anhand der Rollen **Prüfungsamtsmitarbeiter (PA)** und **Studierender** wird dies exemplarisch beschrieben. Jedes Subjekt, das der Rolle **PA** einer Hochschule zugeordnet ist, darf die Noten und abgelegten Leistungen aller Studierenden sehen und bei Bedarf verändern. Ein Subjekt, dem die Rolle **Studierender** zugeordnet*

ist, darf alle eigenen Noten, abgelegte Leistungen und seinen Leitungsnachweis mit Prüfungsverlauf aufrufen, jedoch nicht die seiner Kommilitonen.

Notwendig wird dies, wenn eine große Anzahl von Subjekten mit denselben Zugriffsrechten Zugriff auf Daten mit unterschiedlichen Domänen haben. Werden die Zugriffsrechte mit Hilfe von individuellen Objekten und Operatoren beschrieben, führt dies zu individuellen Rollen. Dies bedeutet, dass für jeden Studierenden eine individuelle Rolle existieren müsste. Mit parametrisierten Rollen bzw. Zugriffsrechten wird versucht, dieses Problem zu lösen, um die Anzahl der Rollen und die Administration überschaubar zu halten (Ge und Osborn 2004, S. 1).

Ein weiteres mögliches Konzept sind Rollenschablonen. Diese erweitern das Konzept der Rolle und kapseln parametrisierbare Zugriffsrechte. Die parametrisierbaren Zugriffsrechte bestehen aus dem Tripel Objekt, Operator und logischer Ausdruck, der beim Zugriff auf das Zielanwendungssystem überprüft wird. Eine Limitation dieses Ansatzes ist, dass eine Rolle nur Zugriffsrechte mit denselben logischen Ausdrücken umfassen kann. Sie unterscheiden zwischen beschränkten und parametrisierbaren Zugriffsrechten. Bei den beschränkten Zugriffsrechten wird der logische Ausdruck auf einen festen vorgegebenen Wert hin überprüft, während bei parametrisierbaren Zugriffsrechten der Vergleichswert eine Variable ist, deren Inhalt erst zur Laufzeit ermittelt wird. Mit den parametrisierbaren Rollen und Zugriffsrechten wird nicht der Funktionsumfang eingeschränkt, sondern ausschließlich die Domäne der Daten (Giuri und Iglio 1997, S. 155–157).

Parametrisierbare Zugriffsrechte

Ein einfaches Zugriffsrecht ist ein Tupel aus Objekt und Operator. Ein parametrisierbares Zugriffsrecht besteht aus Objekt und Operator sowie einer Menge von Parametern. Jeder Parameter wird zerlegt in einen Namen und eine zu überprüfende Domäne - den Ausprägungen des Parameters. Damit ergibt sich, dass sich ein parametrisierbares Zugriffsrecht $\mathbf{Z_P}$ aus Objekt, Operator und einer Parametermenge $\mathbf{P}$ zusammensetzt, wobei $\mathbf{P}$ aus einer Parameterdomäne und Parameterwerten besteht (Ge und Osborn 2004, S. 9).

Parametrisierbare Rolle

Einer Rolle werden sowohl einfache als auch parametrisierbare Zugriffsrechte zugeordnet. Beinhaltet eine Rolle parametrisierbare Zugriffsrechte, so ist diese ebenfalls

parametrisierbar. Eine parametrisierbare Rolle enthält die Parametermengen aller zugeordneten parametrisierbaren Zugriffsrechte. Aus der Rolle **Ro** wird die parametrisierbare Rolle $\mathbf{Ro_P}$, die sich aus der Rolle **Ro** und der Parametermenge **P** ergibt (Ge und Osborn 2004, S. 10).

Die Erweiterung um parametrisierte Rollen und Zugriffsrechte erhöht die Verwendbarkeit des RBAC-Models für Anwendungen, in denen Subjekte ihre privaten Daten sehen können. Es wird damit das Prinzip der minimalen Rechte unterstützt, da Subjekte nur Zugriff auf Daten erhalten, die sie benötigen.

4.8 Strukturkonzepte für Rollen

Zusätzlich zu den beschriebenen Konzepten wird versucht, die Rollen so zu strukturieren, dass eine bessere Zugriffskontrolle gewährleistet und die Administration erleichtert werden kann. Beispiele hierfür sind:

- Rollengruppen (Na und Cheon 2000),
- strukturelle und funktionale Rollen (Coyne und Davis 2008) sowie
- Rollenmapping (Park et al. 2004).

Rollengruppen

Eine Möglichkeit, die Rollen und Rollenhierarchien übersichtlicher zu gestalten, bieten sog. Rollengruppen. Eine Rollengruppe besteht aus Rollen, die keine gemeinsame Hierarchie besitzen. Damit werden die Hierarchien entschlackt und es kommt bei sich ausschließenden Rollen nicht so leicht zu Konflikten, die aufgelöst werden müssen. Im Krankenhaus gibt es z. B. die Rollengruppen: Ärzte, Apotheker und Schwestern (Na und Cheon 2000, S. 41). Jede Rollengruppe hat ihre eigenen Rollenhierarchien.

Strukturelle und funktionale Rollen

Eine weitere Möglichkeit Rollen zu strukturieren, bietet sich mit der Unterteilung in strukturelle und funktionale Rollen. Eine strukturelle Rolle definiert den Eintrittspunkt in ein Anwendungssystem. Eine funktionale Rolle definiert die Zugriffskontrolle innerhalb eines Anwendungssystems. Das bedeutet jedoch, dass es zu jeder strukturellen Rolle eine gespiegelte funktionale Rolle geben muss, denn die funktionale Rolle gewährt nicht den Zugriff zu der Anwendung und ist alleine nutzlos (Coyne und Davis 2008, S. 60-61, 93).

Rollenmapping

Meistens beschäftigt sich die Literatur im Zusammenhang mit RBAC mit dem Fokus auf eine Domäne bzw. mehrere Domänen auf derselben Ebene. RBAC an sich bietet bereits Vorteile bei der Zugriffskontrolle, aber es muss auch eine Betrachtung verschiedener Anwendungssysteme und innerhalb von Organisationen erfolgen. Als Grundlage werden Rollen wie folgt unterteilt:

- Organisatorische Rollen reflektieren einen individuellen Platz in einer organisatorischen Hierarchie.
- Systemrollen basieren auf Funktionen und Aufgaben eines Subjektes in einem Zielanwendungssystem.
- Unternehmensrollen werden benötigt beim Zusammenarbeiten von mehreren Organisationen.

Es werden Rollen auf Ebene der Anwendungssysteme und der Organisation gebildet. In einem ersten Schritt werden die Rollen und deren Einordnung in eine Rollenhierarchie innerhalb der Organisation definiert und dokumentiert. Für jedes Anwendungssystem werden die Systemrollen mit ihren Zugriffsrechten und Rollenhierarchien, die für das entsprechende Anwendungssystem notwendig sind, festgelegt. Um nun diese Systemrollen den organisatorischen Rollen zuzuordnen, wird ein Mapping zwischen diesen beiden Rollentypen vorgenommen. Mit diesem Ansatz soll verhindert werden, dass Subjekte unterschiedliche Rollen in verschiedenen Anwendungssystemen erhalten (Park et al. 2004, S. 168–169).

4.9 Zusammenfassung

Das RBAC-Referenzmodell ist prädestiniert, um darauf aufbauend ein erweitertes rollenbasiertes Zugriffskontrollmodell zu entwickeln. Dazu wurden in diesem Kapitel weiterführende Untersuchungen vorgenommen und Erweiterungen vorgestellt. Es fand eine Abgrenzung des Rollenkonzeptes zum Gruppenkonzept der Zugriffsmatrix statt. Die verschiedenen Beschränkungen der Rollen durch Aufgabentrennung auf Ebene der Rollen und der Subjektzuordnung wurden betrachtet. Nachdem ausgewählte Rollenhierarchien unter verschiedenen Aspekten beleuchtet wurden, wurden Lösungen herausgearbeitet, um Aufgabentrennung und Rollenhierarchien gemeinsam verwenden zu können: Die Rollenhierarchie sollte als eine is_a-Hierarchie modelliert werden und dabei das Konzept der virtuellen Rollen berücksichtigen. Durch eine

getrennte Modellierung von der Zugriffsrechtsvererbung und Aktivierungshierarchie können Kontrollprinzipien und Rollenhierarchie gut miteinander kombiniert werden.

Die Administration mit Hilfe von RBAC in verschiedene Bereiche aufzuteilen und die Delegation als dezentrale, zeitliche Weitergabe von Zugriffsrechten zeigen Möglichkeiten der Flexibilisierung der Zugriffsrechtszuordnungen und Subjektzuordnungen. Ein weiteres Konzept zur Entlastung der Administration ist eine regelbasierte Subjektzuordnung während der Laufzeit. Außerdem besteht die Möglichkeit die ausgelieferten Daten anhand von Domänen zu beschränken. Für beide Konzepte können parametrisierte Zugriffsrechte und Rollen eingesetzt werden und erhöhen damit die Flexibilität der Rollen- und Zugriffsrechtszuordnung. Negative Zugriffsrechte sind zusätzlich eine Möglichkeit, um der Administration die Möglichkeit zu geben, zu verhindern, dass ein Subjekt zu einem bestimmten Zeitpunkt ein Zugriffsrecht erhält, das gegen die unternehmensweite Sicherheitsstrategie verstößt.

Neben der Rollenhierarchie wurden zum Abschluss des Kapitels folgende ausgewählte Rollenkonzepte, mit denen Rollen strukturiert werden können, vorgestellt: Rollengruppen, funktionale und strukturelle Rollen, Systemrollen und organisatorische Rollen.

Ausgewählte hier beschriebene Konzepte erweitern modifiziert zusammen mit Neuerungen das Referenzmodell von RBAC zum erweiterten rollenbasierten Zugriffskontrollmodell eRBAC. eRBAC wird ausführlich in Kapitel 6 beschrieben.

5 Der Begriff Rolle im betrieblichen Informationssystem

Das ausgewählte Zugriffskontrollmodell RBAC (siehe Kapitel 3.3) wurde in Kapitel 4 einschließlich verschiedener Konzepte für Erweiterungen eingehend analysiert. Auch in der Literatur wurde festgestellt, dass das Referenzmodell RBAC für die Autorisierung unterschiedlicher Anwendungssysteme prädestiniert ist (Adams 2006, S. 1) und sich während der letzten 20 Jahre zu einem verbreiteten und ausgereiften Modell für die Konzeption und Entwicklung von Zugriffskontrollsystemen zur Gewährleistung der IT-Sicherheit entwickelte (Coyne und Weil 2008, S. 84; Ahn und Sandhu 2000, S. 208).

In diesem Kapitel wird nun das Konzept der Rolle, das zentrale Element von RBAC, in das Konzept des betrieblichen Informationssystems integriert und die Beziehung Aufgabe, Aufgabenträger und Rolle herausgearbeitet. Zunächst wird der Begriff Rolle mit seinen unterschiedlichen Semantiken beschrieben und in Rollenkonzepten kategorisiert und daraus ein ganzheitliches Rollenkonzept abgeleitet. Anhand der Rollenkonzepte wird die Verwendung der Rolle in ausgewählten Workflow-Systemen untersucht. Nach der Einordnung der Rolle in das IS und die Unternehmensarchitektur nach der SOM-Methodik wird ein Metamodell der Rollenzuordnung vorgestellt.

5.1 Rollenkonzepte

Der Begriff Rolle findet u. a. Verwendung in der Soziologie und Organisationslehre und hier sowohl in der Aufbau- als auch Ablauforganisation. Er bezeichnet in Abhängigkeit von der jeweiligen Forschungsdisziplin unterschiedliche, aber auch sich überschneidende Sachverhalte (Biddle und Thomas 1966, S. 21; Lehmann 1999, S. 316; Crook et al. 2002, S. 11). Das Konzept der Rolle wurde ebenso in verschiedene Disziplinen der Informatik übernommen, z. B. Workflow-Systeme, Softwareentwicklung oder Datenbanken. Es existiert selbst in der Informatik keine Übereinstimmung über die verschiedenen Semantiken des Begriffs Rolle, die alle Nutzungen einschließt (Boella et al. 2007, S. 81; Neumann und Strembeck 2001, S. 58). Disziplinübergreifend existieren verschiedene Interpretationen des Begriffs Rolle, die nicht nur in der Organisationslehre zu einer unzureichenden Klarheit des Begriffs Rolle

führen (Fischer 1992, S. 2224–2225). Eine Systematisierung wird anhand von fünf Rollenkonzepten vorgenommen. Die Einteilung der Rollenkonzepte orientiert sich dabei an (Walther 2005, S. 6–15). Als Einstieg werden ausgewählte Beschreibungen, die die verschiedenen Interpretationen aufzeigen, vorgestellt:

Eine Rolle ist:

- ein Bündel von Verhaltenserwartungen, „die an eine soziale Position gerichtet sind“ (Alisch 2004, S. 2565),
- eine Position innerhalb einer Aufbauorganisation (Beresnevichiene 2003, S. 19),
- als abstrakter Nutzer zu sehen (Lau und Gerhardt 1994, S. 66).

Eine Rolle kann

- Kompetenz repräsentieren, um eine spezielle Aufgabe zu erfüllen (Sandhu et al. 1996, S. 38),
- Autorität und Verantwortung verkörpern (Sandhu et al. 1996, S. 38),
- eine Menge an Transaktionen darstellen, die ausgeführt werden können (Ferraiolo und Kuhn 1992, S. 4),
- Beziehungen zwischen Geschäftspartnern, wie Kunde und Lieferant abbilden (Gebel 2003, S. 8),
- eine Menge von Zugriffsrechten festsetzen (Nyanchama und Osborn 1996, S. 3).

Durch die verschiedenen Disziplinen können folgende Rollenkonzepte gebildet werden:

- verhaltensorientiertes Rollenkonzept
- organisationsorientiertes Rollenkonzept
- aufgabenorientiertes Rollenkonzept
- kompetenzorientiertes Rollenkonzept
- berechtigungsorientiertes Rollenkonzept

Die daraus entstehende Abgrenzung der einzelnen Rollenkonzepte kann nicht als disjunkt betrachtet werden. Es bestehen Überschneidungen in Begrifflichkeiten und Themen. Zum Abschluss dieses Kapitels wird aus den Konzepten ein ganzheitliches Rollenkonzept entwickelt, um die Beziehungen der Rolle verdeutlichen zu können.

5.1.1 Verhaltensorientiertes Rollenkonzept

Eine Rolle im Kontext der Soziologie fasst die Verhaltenserwartungen der Umwelt an eine Position zusammen. Eine Position hat in einem sozialen System Rechte und Pflichten zur Folge (Merton 1957, S. 108; Moffett 1998, S. 63; Thomas und Biddle 1966, S. 4). In der Organisationssoziologie wird unter Rolle die Gesamtheit generalisierter normativer Verhaltenserwartungen verstanden, die an den Inhaber einer bestimmten sozialen Position bzw. Stelle gerichtet sind (Alisch 2004, S. 2565; Mayntz 1980, S. 2044; Rühli 1993, S. 109; Fischer 1992, S. 2224).

Im verhaltensorientierten Rollenkonzept ist eine Rolle ein sozialwissenschaftlicher Begriff zur Kennzeichnung eines Systems von Verhaltensregeln, die meist durch Erwartungen definiert werden. Im Allgemeinen werden diese Erwartungen an den Inhaber einer bestimmten Position herangetragen (Alisch 2004, S. 2565). Die Verhaltens- und Leistungserwartungen werden über die Rolle auf eine Person projiziert. In einem Unternehmen sind Verhaltens- und Leistungserwartungen in einer Stelle gebündelt und richten sich an einen potentiellen Aufgabenträger (Bokranz und Kasten 2001, S. 51; Schreyögg 2008, S. 102). Eine Stelle konkretisiert damit die Rollenerwartungen eines Unternehmens an einen Mitarbeiter (Picot et al. 1997, S. 167).

5.1.2 Organisationsorientiertes Rollenkonzept

Das organisationsorientierte Rollenkonzept stellt die Aufbauorganisation in den Mittelpunkt und betrachtet den Zusammenhang von Stellen und Rollen innerhalb einer Organisation. Aus Sicht der Aufbauorganisation ist eine Stelle die elementarste selbständig handelnde organisatorische Einheit eines sozio-technischen Systems. Eine Stelle ist eine synthetische Zusammenfassung von Teilaufgaben auf einen gedanklich angenommenen Aufgabenträger, der als Erfüllungssubjekt betrachtet wird (Kosiol 1976, S. 89). Sie umfasst damit die Gesamtheit der einem einzelnen Aufgabenträger übertragenen Aufgaben und bezieht sich auf die Kapazität eines Aufgabenträgers (Ferstl und Sinz 2013, S. 109; Galler 1997, S. 50). Eine Stelle wird dabei unabhängig vom konkreten Aufgabenträger als Stelleninhaber definiert. Nach der Zuweisung der Stelle an einen bestimmten Platz innerhalb einer Organisation entsteht eine Position, die mit einem bestimmten Status verbunden ist (Staehle 1999, S. 271).

„Obschon zu jeder Position oder zu jedem Status (mindestens) eine Rolle gehört, ist doch zwischen diesen zu trennen. Erstere bezeichnen - mehr formal - Stellen (Orte) im Gefüge sozialer Interaktion, die von Personen innegehabt, erworben und verloren werden können, während die Rolle angibt, wie sich die jeweiligen Inhaber einer Position verhalten sollen. Die Rolle stellt einen Komplex zusammenhängender Verhaltensweisen dar, die in genereller Form erwartet werden“ (Steinmann und Schreyögg 2000, S. 544). Damit stellt eine Rolle das Insgesamt der Verhaltenserwartungen dar, welche die Organisation und ihre Mitglieder an den Inhaber einer bestimmten Position richten (Staehle 1999, S. 272; Hoffmann 1976, S. 94).

Der Arbeitsumfang einer Stelle wird anhand der Stellenbeschreibung definiert, die oftmals nur Kompetenzen und Verantwortlichkeiten definiert, da diese beständiger sind als einzelne Regelungen (Kieser und Walgenbach 2003, S. 170f). Damit wird in einer Stelle der Zuständigkeits- und Kompetenzbereich eines Aufgabenträgers abgegrenzt. Die Zuordnung von Aufgaben zu Aufgabenträgern als Stelleninhaber ist stets mit der Übertragung von Kompetenz und Verantwortung verbunden (Ferstl und Sinz 2013, S. 78). Es sollte eine Kongruenz zwischen Aufgabe, Verantwortung und Kompetenz bestehen (Bleicher 1980a, S. 1057; Bokranz und Kasten 2001, S. 94). Verantwortung ist dabei die Verpflichtung eines Aufgabenträgers, Rechenschaft abzulegen (Bleicher 1980b, S. 2283) und bedarf dreier Konstruktionsbedingungen: Handlungsziele (Sach- und Formalziele), Handlungsfähigkeit und Handlungsspielraum (Bronner 1992, S. 2504).

„Eine organisatorische Rolle kann auch unabhängig von der Person des möglichen Inhabers definiert werden, und die damit verbundenen Verhaltenserwartungen können sich auf eine ebenso personenunabhängig gebildete Position bzw. Stelle beziehen“ (Staehle 1999, S. 273). Im organisationsorientierten Rollenkonzept werden Stellen mit ähnlichen organisatorischen Charakteristika wie Kompetenzen und Qualifikationen, die sich z. B. auf derselben hierarchischen Ebene oder in derselben organisatorischen Einheit befinden, zu Rollen zusammengefasst (Galler 1997, S. 52; Rupietta und Wernke 1994, S. 143; Walther 2005, S. 11), da an diese Stellen die gleichen Verhaltenserwartungen gestellt werden. Eine Rolle kann beliebig vielen Aufgabenträgern unabhängig voneinander zugewiesen werden, so dass Aufgabenträger mit denselben Rollen in funktionsmäßiger, räumlicher und zeitlicher Art die

Stellvertretung (Blümle 1975, S. 1888) übernehmen können (Reichert und Dadam 2000, S. 27; Walther 2005, S. 11).

5.1.3 Aufgabenorientiertes Rollenkonzept

Die Aufgabe ist der Ausgangspunkt für die organisatorische Gestaltung und steht im Mittelpunkt einer Organisation (Nordsieck 1972, S. 8; Hoffmann 1976, S. 72). Sie wird sowohl in der Aufbauorganisation als auch in der Ablauforganisation betrachtet. Die Ablauforganisation beschreibt die Art der Durchführung, den Arbeitsprozess und die zeitliche und räumliche Koordination der Aufgabendurchführung und sieht die Aufgabe als Teilbereich eines Prozesses an (Nordsieck 1972, S. 10). Die Stellen- und Abteilungsbildung erfolgt dabei unter Berücksichtigung der spezifischen Anforderungen des Ablaufs betrieblicher Prozesse im Rahmen der Leistungserstellung und -verwertung (Gaitanides 1983, S. 62). Aus Sicht der Ablauforganisation wird vom Funktionsbereich des Aufgabenträgers gesprochen, wenn Aufgaben auf einen personellen Aufgabenträger bezogen oder übertragen werden (Kosiol 1976, S. 45). Eine Funktion ist eine Verrichtung (z. B. Einkaufen, Verkaufen, Planen), die zur Erfüllung einer Aufgabe notwendig ist. Die Stelle befindet sich am Übergang zwischen „Aufbau- und Ablauforganisation; an ihr vollzieht sich der Schritt von der Aufgabenzuteilung zur Aufgabenerfüllung“ (Gaitanides 1983, S. 2).

Die Aufgabe wird über die Stelle einem Aufgabenträger zugeordnet (Blohm 1977, S. 52). Funktion und Stelle betrachten beide die Zusammenfassung von Aufgaben auf Aufgabenträger, die Stelle aus Sicht der Aufbauorganisation und die Funktion aus Sicht der Ablauforganisation. „Die Aufbauorganisation definiert den Rahmen, innerhalb dessen sich die Ablauforganisation vollzieht“ (Rupietta und Wernke 1994, S. 135).

Die Gesamtaufgabe wird, wie in **Abb. 5-1** darstellt, durch Analyse in ihre Teilaufgaben zerlegt und durch eine Aufgabensynthese die Aufbauorganisation entwickelt. Die Teilaufgaben werden durch Arbeitsanalyse in Arbeitselemente zerlegt. Durch Arbeitssynthese wird daraus die Ablauforganisation unter Berücksichtigung von Aufgabenträgern, Raum und Zeit gebildet.

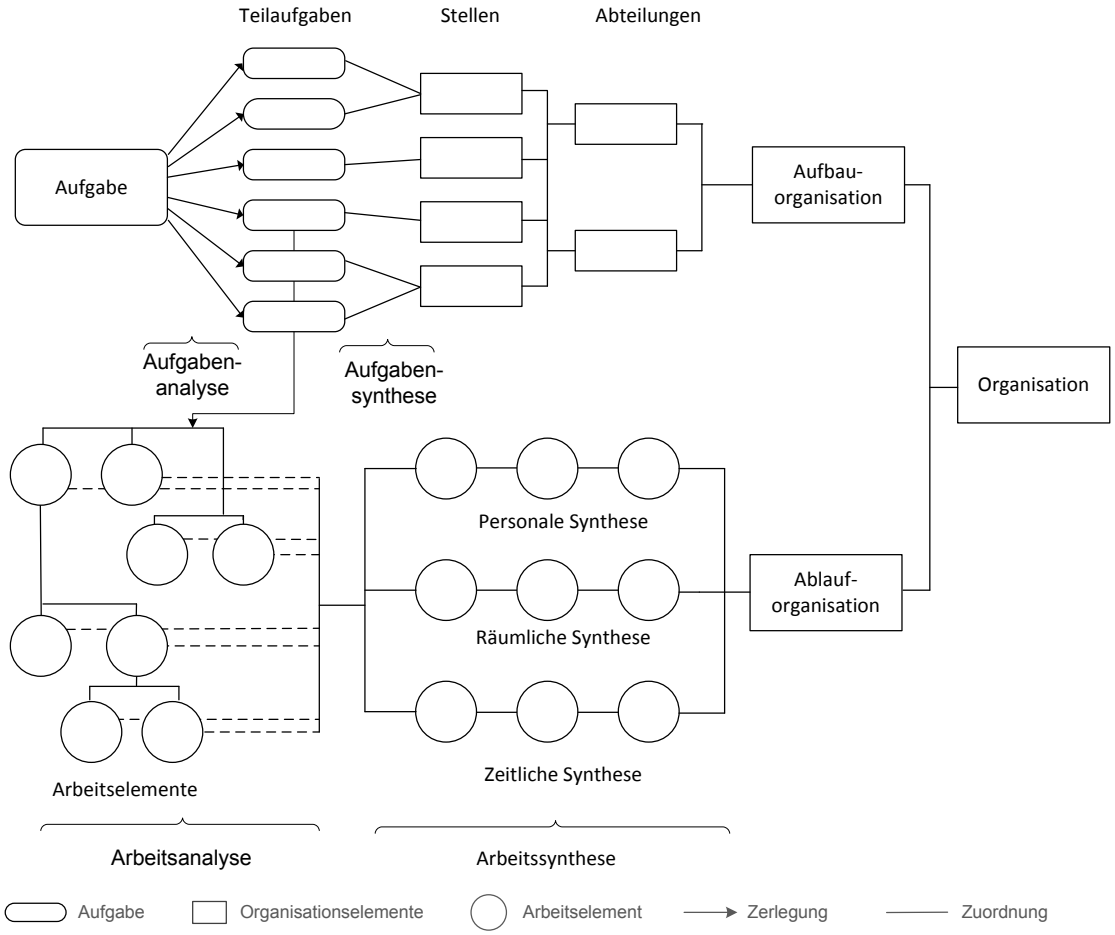

Abb. 5-1 Aufgabenorientiertes Modell der Organisationsgestaltung nach (Schreyögg 2008, S. 105; Frese 1992, S. 250)

Aufgaben werden autorisiert und initiiert in Übereinstimmung mit den Rollen, Verantwortungen und Rechten innerhalb einer Organisation (Thomas und Sandhu 1994, S. 67). Im aufgabenorientierten Rollenkonzept stellt die Rolle eine Zusammenfassung von Aufgaben dar, die einem Aufgabenträger zugeordnet werden können, wobei einem Aufgabenträger auch mehrere Rollen zugeordnet werden können (Esswein 1992, S. 8).

> *Beispielsweise besitzt die Rolle Prüfer u. a. die Aufgaben* ***Noten von Prüfungen erfassen*** *und* ***Lehrveranstaltungen erzeugen****. Die Rollen Studierender, Prüfer und Prüfungsamt können die Aufgabe* ***Student zu einer Prüfung anmelden*** *ausführen.*

Durch dieses Rollenkonzept können Prozess- und Organisationsdefinition voneinander getrennt werden. Wenn ein neuer Mitarbeiter in das Unternehmen eintritt, muss nicht die Prozessdefinition geändert, sondern nur die entsprechende Rolle zugeordnet

werden (Walther 2005, S. 12). Das Lösungsverfahren einer Aufgabe besteht aus einer Menge von Aktionen und dazugehöriger Aktionensteuerung (Ferstl und Sinz 2013, S. 304). Eine Rolle kann daher auch als eine Bündelung von Aktionen angesehen werden.

5.1.4 Kompetenzorientiertes Rollenkonzept

Kompetenzen sind in der Organisationslehre die einem Aufgabenträger über eine Stelle übertragenen Rechte, Zuständigkeiten und Befugnisse, die es ihm ermöglichen sollen, innerhalb eines vorgegebenen Handlungsspielraums seine Aufgaben zu erfüllen (Hahn 1975, S. 1111). Befugnisse umfassen sowohl die Zuständigkeit für Handlungen, den Handlungsspielraum als auch die Durchsetzbarkeit von Handlungen (Bronner 1992, S. 2507).

Im kompetenzorientierten Rollenkonzept werden Qualitätsanforderungen an eine Rolle definiert, die sich auf folgende vier Kompetenzarten beziehen:

- Fachkompetenz,
- Methodenkompetenz,
- Sozialkompetenz,
- Medienkompetenz.

Die **Fachkompetenz** beschreibt das fach-, prozess-, aufgaben-, organisations- und arbeitsplatzspezifische und fachübergreifende berufliche Wissen, Fertigkeiten und Fähigkeiten, um Aufgaben zu lösen (Frings und Weisbecker 1998, S. 20; Strasmann 1996, S. 13).

Die **Methodenkompetenz** entspricht der Fähigkeit und Bereitschaft, Fachwissen anzuwenden, zu kombinieren und zu ergänzen. „Sie beinhaltet auch die Entwicklung von Abstraktionsfähigkeit, Systemdenken, Planungsfähigkeit und Problemlösungs- sowie Entscheidungsfähigkeiten als auch die Beherrschung von Arbeitsmethoden“ (Frings und Weisbecker 1998, S. 20), um Aufgaben selbständig und systematisch Lösungswege zu finden (Strasmann 1996, S. 13f).

Sozialkompetenz umfasst persönliche Ausprägungen im Umgang mit der eigenen Person (Individualkompetenz) und mit anderen Personen. Individualkompetenz umfasst die Fähigkeit und Bereitschaft, sich selbst im Rahmen der Arbeitsaufgabe oder der Arbeitsgruppe zu entwickeln, eigene Begabung, Motivation, Leistungsbereit-

schaft und die Fähigkeit zur Selbsterkenntnis zu entfalten. Die Fähigkeit zur Selbsterkenntnis umfasst die Wahrnehmung und Beurteilung der eigenen Wirkung und Verhaltensweisen, z. B. Selbständigkeit, Zielstrebigkeit, Zuverlässigkeit, Flexibilität, Durchhaltevermögen und Kreativität (Frings und Weisbecker 1998, S. 20; Strasmann 1996, S. 14). Im Umgang mit anderen Personen ist dies die Fähigkeit und Bereitschaft, sich mit anderen, unabhängig von Alter, Herkunft und Bildung verantwortungsbewusst auseinanderzusetzen und sich gruppen- und beziehungsorientiert zu verhalten, z. B. Team-, Kooperations- und Kommunikationsfähigkeit sowie die Fähigkeit zur Delegation.

Die **Medienkompetenz** beinhaltet die Beschaffung, Aufbereitung, Präsentation und Darstellung von Informationen, das Verwalten von Wissen und das ökonomische Filtern von Informationen nach deren Wichtigkeit sowie die Beherrschung verschiedener Medien. Durch die Verbreitung des Internets und der damit verbundenen Informationsexplosion wird die Medienkompetenz immer wichtiger (Frings und Weisbecker 1998, S. 20; Klippert 2000, S. 30).

Im kompetenzorientierten Rollenkonzept beschreibt die Rolle eine Grundmenge von Qualitätsanforderungen über Fähigkeiten, Erfahrungen, Kenntnisse und Kompetenzen, die eine Person besitzen muss, um bestimmte Aufgaben durchzuführen (van Aalst der und Hee van 2002, S. 353; Frings und Weisbecker 1998, S. 19f; Graf 2002, S. 48). Es wird nichts über die Person ausgesagt, die diese Rolle einnimmt. Eine Rolle wird mit Aufgaben und Verantwortlichkeiten verknüpft. Sie kann für mehrere Aufgaben zuständig sein, aber umgekehrt auch mehrere Rollen für eine Aufgabe (Frings und Weisbecker 1998, S. 19).

Alle Aufgabenträger, die über ihre Rolle für die Bearbeitung einer bestimmten Aufgabe qualifiziert sind, können diese ausführen und sind damit austauschbar (Reichert und Dadam 2000, S. 26). Auch in Workflow-Systemen wird bei der Abbildung der Aufbauorganisation der Begriff Rolle verwendet, um die für eine Ausführung notwendigen Qualifikationen zu bündeln und Kompetenzen zu beschreiben, die einem Aufgabenträger übertragen werden (Rosemann und Mühlen 1997, S. 2).

5.1.5 Berechtigungsorientiertes Rollenkonzept

Im berechtigungsorientierten Rollenkonzept steht die Vergabe von Zugriffsrechten im Fokus. Die Rolle ist hierbei eine Kapselung von Berechtigungen, Verantwortlich-

keiten und Verpflichtungen innerhalb eines IS (Nyanchama und Osborn 1996, S. 131).

Der Rollenbegriff im rollenbasierten Zugriffskontrollmodell (siehe Kapitel 3.2.7) wird als semantisches Konstrukt angesehen, das Zugriffsfunktionalität formuliert. Eine Rolle ist eine Arbeitsfunktion innerhalb des Kontextes einer Organisation. Sie ist verknüpft mit Semantiken, welche Autorität und Verantwortung berücksichtigen, die auf den Aufgabenträger übertragen werden, dem eine Rolle zugeordnet ist. Rollen verbinden eine Menge von Aufgabenträgern mit einer Menge von Zugriffsrechten (ANSI INCITS 359-2004 2004, S. 3). Eine Rolle kann als eine Menge von Zugriffsrechten auf Funktionen betrachtet werden, die ein oder mehrere Aufgabenträger innerhalb eines Kontextes ausführen dürfen. Rollen dienen der Abstraktion, um Zugriffsrechte zu verwalten. Anstatt Aufgabenträgern Zugriffsrechte direkt zuzuweisen, wird eine Rolle definiert, die mehreren Aufgabenträgern zugeordnet werden kann. Die Zugriffsentscheidung wird anhand der zugeordneten Zugriffsrechte einer Rolle, die einem individuellen Aufgabenträger als Teil der Organisation übertragen wurde, getroffen.

5.1.6 Ganzheitliches Rollenkonzept

In diesem Kapitel werden die einzelnen Rollenkonzepte wie **Abb. 5-2** zeigt zusammengeführt. Aufgaben werden über Stellen personellen Aufgabenträgern zugeordnet. Eine Rolle beschreibt die notwendigen Kompetenzen sowie die damit verbundene Verantwortung zur Erledigung einer Aufgabe. Rollen kapseln zudem Zugriffsrechte. Über die Rolle erhält ein personeller Aufgabenträger die notwendigen Zugriffsrechte, um seine Aufgaben, die ihm über die Stelle zugeordnet sind, erledigen zu können.

Eine Stelle erzeugt bei der Zuordnung zu einem Aufgabenträger Verantwortung. Diese wird durch die Rolle an den zugeordneten Aufgabenträger übergeben. Ein personeller Aufgabenträger befindet sich in einer bestimmten Position. Diese Position bestimmt die Verhaltens- bzw. Leistungserwartungen, die über die Rolle an einen Aufgabenträger gerichtet wird.

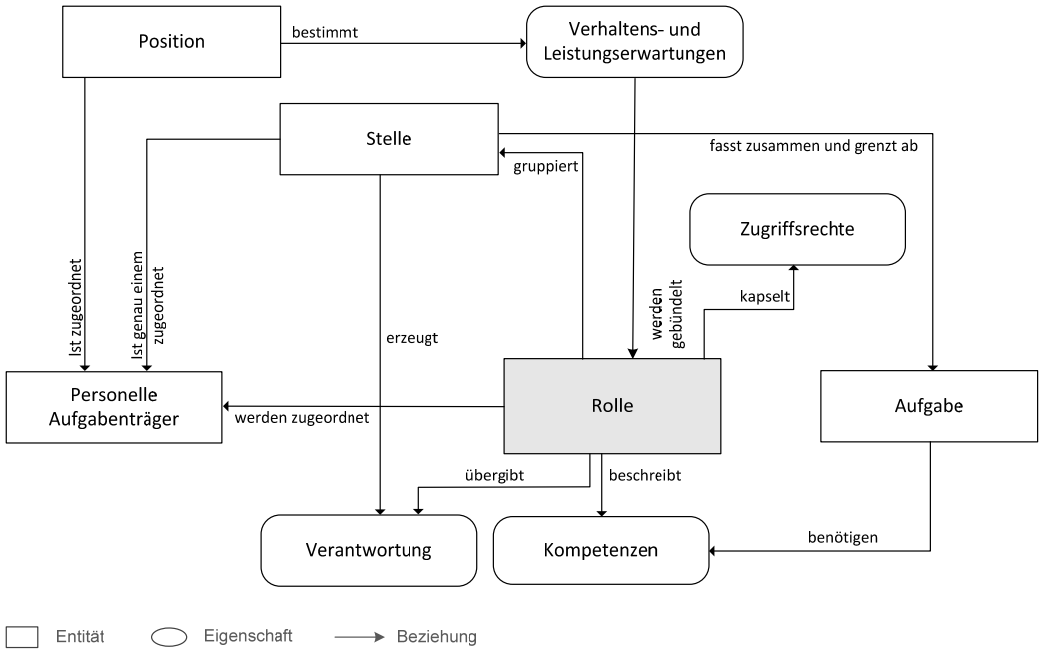

Abb. 5-2 Zusammenführung der Rollenkonzepte

Das ganzheitliche Rollenkonzept zeigt, dass die Rolle nicht ausschließlich Zugriffsrechte kapselt, sondern im Gesamtkontext von Stelle, Position und Aufgabe zu sehen ist. Eine Rollenzuordnung ist zudem nichts Statisches – so kann ein Aufgabenträger im Laufe seines Arbeitslebens unterschiedliche Rollen einnehmen, aber auch zum selben Zeitpunkt mehrere Rollen innehaben (van Aalst der und Hee van 2002, S. 15).

5.2 Rollenkonzepte in ausgewählten Workflow-Systemen

Im Folgenden werden ausgewählte Workflow-Systeme und Forschungsarbeiten in Bezug auf die Implementierung und Umsetzung der dort eingesetzten Rollenkonzepte hin analysiert. Dabei werden nicht Workflow-Systeme als Ganzes, sondern der Teilbereich untersucht, der sich mit der Abbildung bzw. Modellierung der Organisationsstruktur und Rollen beschäftigt. Aus der Vielzahl von Workflow-Systemen werden exemplarisch **WorkParty**, **SAP R/3®**, **ProMInanD**, **SAP Business Workflow®**, die einschlägige Literatur, das von (Galler 1995) erarbeitete **Metamodell** des Workflow-Managements sowie die Beschreibung des Workflow Management Coalition betrachtet. Im Bereich der Workflow-Systeme wird Rolle vielfältig und facettenreich verwendet (Lehmann 1999, S. 316) und es wird von einem Homonymkonflikt gesprochen (Mühlen und Rosemann 1996, S. 14). Die Umsetzungsmöglichkeiten von organisatorischen Konstrukten in Workflow-Systemen unterscheiden sich teilweise erheblich. Üblich ist die Verwendung von Organi-

sationseinheiten, z. B. im Sinne von Abteilungen, die direkte Adressierung von Mitarbeitern sowie die Referenzierung von Rollen (Müller und Stolp 1999, S. 141).

Im Organisations- und Ressourcenmanagement (ORM), das vom Workflow-System **WorkParty** genutzt wird, besteht die Möglichkeit, die Aufbauorganisation, anhand von Rollen zu hinterlegen, um eine Entscheidung treffen zu können, wer den nächsten Arbeitsschritt ausführen darf. Rollen definieren dabei gemeinsame Eigenschaften und Kompetenzen und beziehen die gesamte Organisation mit ein (Jablonski 1995, S. 87; Rupietta und Wernke 1994, S. 142–144). Zur Laufzeit werden anhand der zugeordneten Rollen die möglichen ausführenden Aufgabenträger ermittelt. Rollen bündeln hier die für eine Ausführung notwendigen Qualifikationen und beschreiben die Kompetenzen, die einem Rollenträger übertragen werden. Es wird das kompetenzorientierte Rollenkonzept angewendet. Eine Rolle repräsentiert zum einen die für die Ausführung einer Aktivität notwendige minimale Qualifikation, zum anderen beschreibt eine Rolle Kompetenzen und Befugnisse, welche einem Rollenträger übertragen werden (Jablonski et al. 1997, S. 102–103; Rosemann und Mühlen 1997, S. 108; Weske 1999, S. 43).

Als weitere Beispiele werden **SAP R/3** und **ProMInanD** untersucht. Die Abbildung der Aufbauorganisation ist ein Teil der SAP R/3-Komponente *PD* (Organisation und Planung). Normalerweise wird eine Aufgabe nicht direkt einem personellen Aufgabenträger zugeordnet, sondern es werden Tätigkeitsprofile festgelegt. Die Rollenauflösung findet zur Laufzeit statt (Kurz 1998, S. 43). In ProMInanD werden Bearbeiter durch ihre Funktion innerhalb einer Organisation (ihrer Rolle) bestimmt. „Stellen und organisatorische Funktionen werden […] Rollen genannt“ (Karbe 1994, S. 125–127). Die Rolle wird in beiden Workflow-Systemen nicht zur Rechteprüfung, sondern zum Abbilden der Aufbauorganisation herangezogen (Becker 1998, S. 196) und entspricht damit dem organisationsorientierten Rollenkonzept.

Im **SAP Business Workflow**® bezeichnet eine Rolle eine Regel, um zur Laufzeit den Aufgabenträger eines sog. Workitems zu ermitteln, wenn die Menge der möglichen Aufgabenträger zu groß und unspezifisch ist. Ein Workitem ist hierbei ein Schritt in der Workflow-Definition oder eine Aufgabe. Durch die Zuordnung werden Zuständigkeiten und Berechtigungen geregelt (SAP 2007). Dabei findet in erster Linie das kompetenzorientierte Rollenkonzept seine Anwendung.

SAP R/3 hatte zunächst ein Berechtigungskonzept, basierend auf der Definition von Benutzertypen wie z. B. Web-User oder R/3-Nutzer und zugeordneten Berechtigungsprofilen. Innerhalb des Workflows müssen den Aufgabenträgern, je nach Aufgaben, noch Zugriffsrechte zu ihrem Berechtigungsprofil hinzugefügt oder entzogen werden. Der Begriff Rolle findet innerhalb des Rechtekonzeptes keine Verwendung, die Rechteprüfung und -verwaltung ist mit dem Gruppenkonzept der Zugriffsmatrix (siehe Kapitel 3.2.1) vergleichbar.

In den neueren Versionen von **SAP R/3** werden Zugriffsrechte über Rollen definiert und das berechtigungsorientierte Rollenkonzept angewendet. Rollen sind die wichtigste Berechtigungskomponente. Es existieren sowohl Einzelrollen als auch zusammengesetzte Rollen. Eine Rolle besteht aus Berechtigungsobjekten, Berechtigungen und Berechtigungsprofilen. Berechtigungen sind hier keine Zugriffsrechte, sondern eine technische Komponente innerhalb der Rolle. Berechtigungsobjekte dienen als Vorlage für die Verwendung im Programmcode und werden zu Berechtigungen zusammengefasst. Diese werden einem Berechtigungsprofil in der Rolle zugeordnet. Eine Rolle wird anschließend einem Subjekt zugeordnet. Das R/3-System zieht beim Anmelden die Berechtigungen aus dem Benutzerstamm in den Benutzerspeicher und prüft alle Zugriffsrechte gegen diesen Speicher[50] (Linkies und Off 2006, S. 205–213).

Eine weitere Beschreibung der Rolle in Workflow-Systemen findet sich in (Jablonski et al. 1997, S. 488). „Eine Rolle wird durch eine Menge von Zuständigkeiten, die eine Person durch Stellenzuordnung (Stellenbesetzung) erwirbt, charakterisiert. Zuständigkeit ist dabei ein Oberbegriff für Funktionen (Ausführungshandlungen), Verantwortungsbereiche (Entscheidungshoheit über Personen, Arbeitsmittel, Arbeitsstoffe, Termine, Finanzmittel) und Ziele (Resultate)“ (Jablonski et al. 1997, S. 488). Auch diese Definition beschreibt die Rolle im Sinne eines organisationsorientierten Rollenkonzeptes.

Auch das aufgabenorientierte Rollenkonzept findet Anwendung, wie die beiden folgenden Beschreibungen von Rollen in Workflow-Systemen belegen:

- Um für einen Arbeitsschritt während des Arbeitsablaufs geeignete Aufgabenträger aus der Organisationsstruktur zu ermitteln stehen Zuweisungsregeln zur Verfügung. Diese Zuweisungsregeln können einer Rolle, einer Organisation oder

[50] Eine ausführliche Beschreibung des Rollenkonzeptes findet sich in Linkies und Off (2006).

direkt einem Aufgabenträger zugeordnet werden. Es kann einem Aufgabenträger eine oder mehrere Rollen zugeordnet sein. Allen Aufgabenträgern, die die Zuweisungsregeln erfüllen, wird der Arbeitsschritt zugewiesen (Bussler 1998, S. 10).

- Das Workflow Referenzmodell des WfMC[51] erlaubt es, dass eine Aufgabe von mehreren Mitarbeitern ausgeführt werden kann. Diese Aufgabe wird einer Rolle zugeordnet und erst zur Laufzeit wird der zuständige personelle Aufgabenträger ermittelt (Hollingsworth 2004, S. 304).

In den vorgestellten Workflow-Systemen bzw. Beschreibungen werden vier der vorgestellten Rollenkonzepte verwendet: das organisationsorientierte, aufgabenorientierte, kompetenzorientierte und das berechtigungsorientierte Rollenkonzept. (Galler 1995) beschreibt drei der vorgestellten Rollenkonzepte und zeigt die Zusammenhänge in einem Metamodell der Workflow-Systeme. Rollen finden in Workflow-Systemen unterschiedliche Verwendung, einmal indem Stellen zusammengefasst werden (organisationsorientiertes Rollenkonzept), um Kompetenzen zu definieren (kompetenzorientiertes Rollenkonzept) und einmal als Aggregation von Funktionen im Sinne von Aufgaben (aufgabenorientiertes Rollenkonzept) (Galler 1995, S. 7).

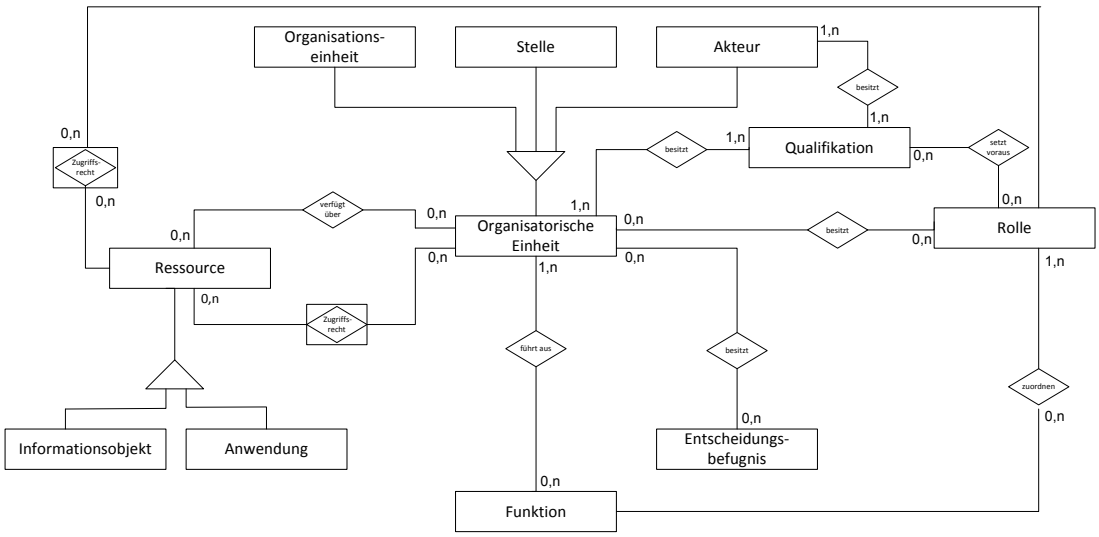

Abb. 5-3 Ausschnitt der Rolle im Metamodell von Workflow-Systemen nach (Galler 1995, S. 30)

Abb. 5-3 zeigt einen Auszug dieses Metamodells in Bezug auf die Rolle. Es werden Funktionen als Tätigkeiten definiert, die durch ein Workflow-System gesteuert werden (Galler 1995, S. 12). Eine organisatorische Einheit kann eine Stelle, Organisati-

[51] Tiefgehende Informationen und Definitionen finden sich unter http://www.wfmc.org/ und WfMC (1999)

onseinheit oder ein Akteur sein, welcher Entscheidungsbefugnis besitzt. Ein Akteur ist dabei ein personeller Aufgabenträger und führt entweder direkt oder indirekt über Rollen Aufgaben aus. Akteure besitzen Qualifikationen. Die Begriffe Stelle und Organisationseinheit werden, wie unter Kapitel 5.1.2 beschrieben, verwendet. Informationseinheiten und Anwendungssysteme sind Ressourcen, die mittels Zugriffsrechten organisatorischen Einheiten bzw. Rollen zugeordnet werden.

Der Workflow Management Coalition bezeichnet eine Rolle oder eine Ressource als abstrakter Aufgabenträger (**abstract actor**). Während der Laufzeit werden diese abstrakten Definitionen ausgewertet und konkreten Aufgabenträgern zugeordnet (WfMC 2005, S. 99). Dabei wird zwischen einer organisatorischen Rolle und einer Prozessrolle unterschieden. Die organisatorische Rolle stellt einen Teilnehmerkreis dar, der eine spezifische Menge von Attributen wie Qualifikationen und Fähigkeiten besitzt. Die Prozessrolle definiert dazu im Gegensatz den Kontext eines Aufgabenträgers in einem speziellen Prozess. Sie umfasst organisatorische Aspekte wie Struktur und Beziehung sowie Verantwortung und Autorität, bezieht aber auch Attribute wie Fähigkeiten, Ort oder Zeit mit ein (WfMC 2005, S. 100; WfMC 1999, S. 53–54).

Zusammenfassung

Der Fokus bei der Modellierung der Workflow-Systeme liegt naturgemäß auf der Abbildung der Ablauforganisation. Mit Workflow-Systemen wird die Aufgabendurchführung unterstützt und damit die Ablauforganisation abgebildet. Die Modellierung der Aufbauorganisation steht im Kontext des Workflow-Managements zumeist hinter der Analyse der Modellierung der Ablauforganisation zurück (Kirn und Kümmering 1997, S. 59; Rosemann und Mühlen 1997, S. 100; zur Muehlen 2004, S. 271). Es gibt keinen ähnlich umfassenden Ansatz für die Abbildung der Aufbauorganisation und nur begrenzte Möglichkeiten, diese abzubilden (Heilmann 1996, S. 154). Die Rolle kann als Bindeglied zwischen dem Anwendungssystem, dem Workflow und der Aufbauorganisation gesehen werden. Rollen konkretisieren damit das Tätigkeitsprofil des Aufgabenträgers (Janetzke 2001, S. 183).

In Workflow-Systemen wurde die Rolle lange Zeit nicht zur Rechteprüfung verwendet, sondern um eine flexible Zuordnung zur Laufzeit von Aufgabenträgern zu Aufgaben zu ermöglichen (Müller und Stolp 1999, S. 64). Schon früh wurde gefordert,

dass jede Aktion, die in einem Workflow-Prozess ausgeführt werden soll, einer Rolle, die Zugriffsrechte kapselt, zugeordnet wird, um die Komplexität der Sicherheitsadministration zu vereinfachen. Rollen werden anschließend Aufgabenträgern zugeordnet (Bertino et al. 1997a, S. 1). Eine Verwendung neben der Zuordnung zwischen Aufgabe und Aufgabenträger, auch als Konzept zur Zugriffskontrolle, findet sich erst in neueren Versionen von Workflow-Systemen, wie **SAP R/3** zeigt. Die Verwendung der Rolle für beide Aspekte zeigt die folgende Beschreibung: Aus Prozesssicht repräsentiert eine Rolle sowohl Fähigkeiten als auch Zugriffsrechte, die für eine Ausführung einer Aktion im Workflow-System benötigt werden (zur Muehlen 2004, S. 279–281).

5.3 Konzept des betrieblichen Informationssystems

Informationssysteme sind Systeme, die Informationen verarbeiten also z. B. erfassen, speichern und bereitstellen. Der Einsatz findet in Organisationen von Wirtschaft und Verwaltung und auch organisationsübergreifend statt. Diese Informationssysteme werden als betriebliche Informationssysteme bezeichnet (Ferstl und Sinz 2013, S. 3). Für die spätere Einordnung des Begriffs Rolle wird das IS anhand der Aufgaben- und der Aufgabenträgerebene (siehe Kapitel. 2.1.4) untersucht.

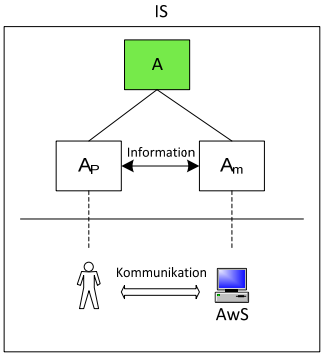

Abb. 5-4 Aufgaben- und Aufgabenträgerebene eines IS

Ein IS enthält eine Menge von Informationsverarbeitungsaufgaben, die durch Informationsbeziehungen verbunden sind. Dies gilt auch für teilautomatisierte Aufgaben. Ein IS enthält daneben eine Menge von Aufgabenträgern (AT)[52], die durch Kommunikationssysteme verbunden sind. Dieser Zusammenhang wird in **Abb. 5-4** schematisch dargestellt.

[52] Eine ausführliche Beschreibung des Begriffs Aufgabenträger findet sich u. a. in Schwarz (1980).

Die Menge der Aufgaben bilden die Aufgabenebene und „die Menge aller Aufgabenträger bildet die Aufgabenträgerebene eines IS“ (Ferstl und Sinz 2013, S. 5). Typischerweise ist bei der Durchführung betrieblicher teilautomatisierter Aufgaben eine Kooperation maschineller und personeller Aufgabenträger notwendig (Ferstl und Sinz 2013, S. 58). „Eine Informationsbeziehung zwischen zwei Aufgaben mit unterschiedlichen Aufgabenträgern wird durch einen Kommunikationskanal zwischen den Aufgabenträgern realisiert.“ (Ferstl und Sinz 2013, S. 5).

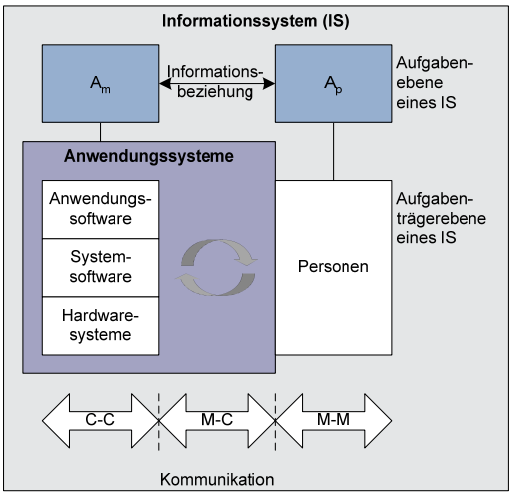

Abb. 5-5 Informationsbeziehungen und Kommunikationssysteme im IS (Ferstl und Sinz 2013, S. 5)

Diese Informationsbeziehung sowie die Kommunikationssysteme sind in **Abb. 5-5** dargestellt. AwS sind Systeme, die auf einzelne automatisierbare Aufgabenbereiche eines IS zugeschnitten sind. Die Anwendungssysteme bestehen aus einer Anwendungssoftware. Diese setzt auf einer Basismaschine, der Systemsoftware, und diese wiederum auf Hardwaresystemen auf (Ferstl und Sinz 2013, S. 5-6, 10).

Eine Aufgabe wird allgemein als Zielsetzung für zweckbezogenes menschliches Handeln definiert (Kosiol 1976, S. 43). Die Außensicht einer Aufgabe, siehe **Abb. 5-6**, definiert das Aufgabenobjekt, die Sach- und Formalziele sowie die Vorereignisse, die eine Aufgabe auslösen und die Nachereignisse, die aus einer Aufgabendurchführung resultieren (Ferstl und Sinz 2013, S. 98). „Die Innensicht einer Aufgabe definiert das Lösungsverfahren (den Verrichtungsvorgang) der Aufgabe“ (Ferstl und Sinz 2013, S. 98). Ein Lösungsverfahren besteht aus einer Menge von Aktionen, die sequentiell oder parallel auf das Aufgabenobjekt einwirken oder Zustände des Aufgabenobjektes erfassen. Das Lösungsverfahren nimmt dabei Bezug auf den Aufgabenträger. Bei personellen Aufgabenträgern wird das Lösungsverfahren meist natürlich sprachlich beschrieben. Für maschinelle Aufgabenträger wird für

die Beschreibung des Lösungsverfahrens eine Programmiersprache verwendet (Ferstl und Sinz 2013, S. 61-64; 98).

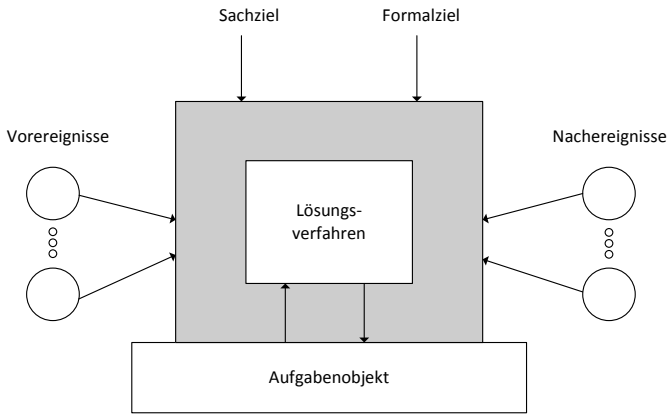

Abb. 5-6 Aufgabenstruktur (Ferstl und Sinz 2013, S. 98)

Ein Aufgabenkomplex für einen personellen Aufgabenträger wird einer Stelle zugeordnet. Die Stellenbeschreibung enthält meist nur die Außensicht der durchzuführenden Aufgaben, die zugehörigen Lösungsverfahren werden vom Stelleninhaber generiert. „Die Verantwortung des Stelleninhabers besteht in erster Linie darin, die Zielerreichung anhand der von ihm gewählten Lösungsverfahren nachzuweisen. Bei der Wahl des Lösungsverfahrens orientiert sich der Stelleninhaber an der ihm zugeteilten Kompetenz. Die Beschreibung einer automatisierbaren Aufgabe ($A_{m)}$ in Form eines Programms beinhaltet die Außen- und Innensicht der Aufgabe“ (Ferstl und Sinz 2013, S. 109). „Die Wahl des Lösungsverfahrens erfolgt zum Zeitpunkt der Definition der automatisierbaren Aufgabe“, also der Erstellung des Anwendungssystems (Ferstl und Sinz 2013, S. 109).

5.4 Rolle und Aufgabenträger im Konzept der Virtualisierung

In diesem Kapitel werden mit Hilfe der vorgestellten Rollenkonzepte, der Untersuchungen in Workflow-Systemen sowie des Konzeptes des IS die Rolle und der Aufgabenträger in Bezug auf Virtualisierung untersucht. Dabei soll analysiert werden, ob die Rolle der Aufgaben- oder Aufgabenträgerebene des IS zugeordnet werden kann und inwieweit sich das Konzept der Virtualisierung auf das Konzept der Rolle anwenden lässt. „Das aus der Informatik stammende Architekturkonzept der Virtualisierung lässt sich in seinen Charakteristika und Realisierungsprinzipien auch auf die Architektur von Unternehmensorganisationen gewinnbringend übertragen“ (Reich-

wald 2000, S. 258). Das Konzept der Virtualisierung im Kontext der Speicherarchitekturen bildet das am besten geeignete Vorbild für die Architektur virtueller Organisationen (Picot et al. 2003, S. 419) und wird dieser Untersuchung zu Grunde gelegt.

Virtualität bezieht sich grundsätzlich immer auf ein Objekt. Dieses virtuelle Objekt besitzt, obwohl es nicht tatsächlich existiert, volle Funktionalität und kann von außen behandelt werden als sei es ein reales Objekt (Picot et al. 2003, S. 164).

> *Beispielsweise besitzt der virtuelle Arbeitsspeicher alle Eigenschaften des realen Arbeitsspeichers, ist aber größer als der reale.*

Im **verhaltensorientierten Rollenkonzept** bündelt die Rolle die Erwartungen an den Inhaber einer bestimmten Position. Die Verhaltensregeln und Erwartungen werden über die Rolle auf eine Person bzw. den potentiellen Aufgabenträger projiziert. Die Rolle ist hier vom Typ Aufgabenträger und erfüllt die Merkmale eines virtuellen Aufgabenträgers: Sie erhält alle Erwartungen, die an einen Aufgabenträger gestellt werden, und ist der Platzhalter für eine reale Person, die eine bestimmte Position einnehmen kann.

Im **organisationsorientierten Rollenkonzept** stellt eine Rolle die Zusammenfassung von Stellen mit gleichartiger Kompetenz und Verantwortlichkeiten dar (Galler 1997, S. 52) und definiert die Verhaltenserwartungen der Organisation an den Rolleninhaber. Die Rolle kann vielen Aufgabenträgern unabhängig voneinander zugewiesen werden, so dass Aufgabenträger mit denselben Rollen in funktionsmäßiger, räumlicher und zeitlicher Art die Stellvertretung übernehmen können (Blümle 1975, S. 1888; Walther 2005, S. 11). Die Rolle ist hier ein virtueller Aufgabenträger und besitzt Verantwortlichkeiten und Verhaltenserwartungen wie ein realer Aufgabenträger ohne Kapazitätsbegrenzung u. a. auf Raum und Zeit des realen Aufgabenträgers.

Im **aufgabenorientierten Rollenkonzept** ist die Rolle eine Zusammenfassung von Aufgaben, die einem Aufgabenträger übertragen werden. Alle einer Rolle zugeordneten Aufgabenträger können diese Aufgaben erfüllen. Die Rolle als virtueller Aufgabenträger bündelt die Aufgaben, die ein zugeordneter Aufgabenträger ausführen soll. Beispielsweise wird damit in Workflow-Systemen erst zur Laufzeit entschieden, wer die anstehende Aufgabe tatsächlich ausführt.

Im **kompetenzorientierten Rollenkonzept** beschreibt eine Rolle eine Grundmenge von Qualitätsanforderungen, um eine bestimmte Aufgabe zu erfüllen. Alle Aufgabenträger, die derselben Rolle zugeordnet sind, sind zur Bearbeitung einer bestimmten Aufgabe qualifiziert, können diese ausführen und ggf. die Stellvertretung übernehmen. Eine Rolle besitzt alle Kompetenzen des realen Aufgabenträgers, jedoch ohne die Kapazitätsbeschränkung des Aufgabenträgers. Auch im kompetenzorientierten Rollenkonzept ist die Rolle ein virtueller Aufgabenträger, da sie alle Qualitätsanforderungen unabhängig von einem konkreten Aufgabenträger bündelt.

Analog kapselt im **berechtigungsorientierten Rollenkonzept** eine Rolle Zugriffsrechte. Die Rolle existiert unabhängig von Subjekten und zieht keine Charakteristiken von Subjekten in Betracht (Kern et al. 2004, S. 90). Als virtueller Aufgabenträger besitzt die Rolle alle Zugriffsrechte der realen Aufgabenträger, die dieser zugeordnet sind. Die Zugriffsrechte stehen einem Aufgabenträger in einer Sitzung zur Erledigung seiner Aufgaben zur Verfügung.

Die Rolle kann in allen Rollenkonzepten als virtueller Aufgabenträger interpretiert werden. Damit dient die Rolle als Virtualisierung eines Aufgabenträgers analog virtueller Betriebsmittel der Komplexitätsreduzierung, Portierbarkeit und Standardisierung (Ferstl und Sinz 2013, S. 373). Auch in Workflow-Systemen werden Rollen als virtuelle organisatorische Objekte (Jablonski 1995, S. 52) oder als abstrakter Aufgabenträger bezeichnet (WfMC 2005, S. 99). Die Rolle als virtueller Aufgabenträger kann also als Repräsentation eines konkreten Aufgabenträgers gesehen werden. Dadurch wird deutlich, dass die Rolle nicht der konkreten Aufgabenträgerebene zugeordnet werden kann, so dass eine zusätzliche Ebene am Übergang vom Geschäftsprozessmodell zur Aufgabenträgerebene und bei der Zuordnung von Aufgaben zu Aufgabenträgern nötig ist.

5.5 Einordnung der Rolle ins betriebliche Informationssystem

Zunächst wird in diesem Kapitel das erarbeitete ganzheitliche Rollenkonzept (siehe **Abb. 5-2**) um Anwendungssysteme und deren Funktionen erweitert. Danach erfolgt eine Untersuchung der Beziehung von Rolle und Zugriffsrechten innerhalb des AwS anhand des ADK-Modells. Das ADK-Modell besteht aus den drei Teilsystemen Anwendungsfunktionen (A), Datenverwaltung (D) und Kommunikation (K). Abschlie-

ßend kann durch eine Einordnung der Rolle, Authentifizierung und Zugriffskontrolle in die Informationsbeziehungen und Kommunikationssysteme des IS gezeigt werden, dass die Rolle ein geeignetes Konzept ist, um Zugriffsrechte innerhalb einer Anwendungssoftware zu bündeln und damit die notwendigen Funktionen innerhalb dieser freizugeben, um Aufgaben zu erfüllen.

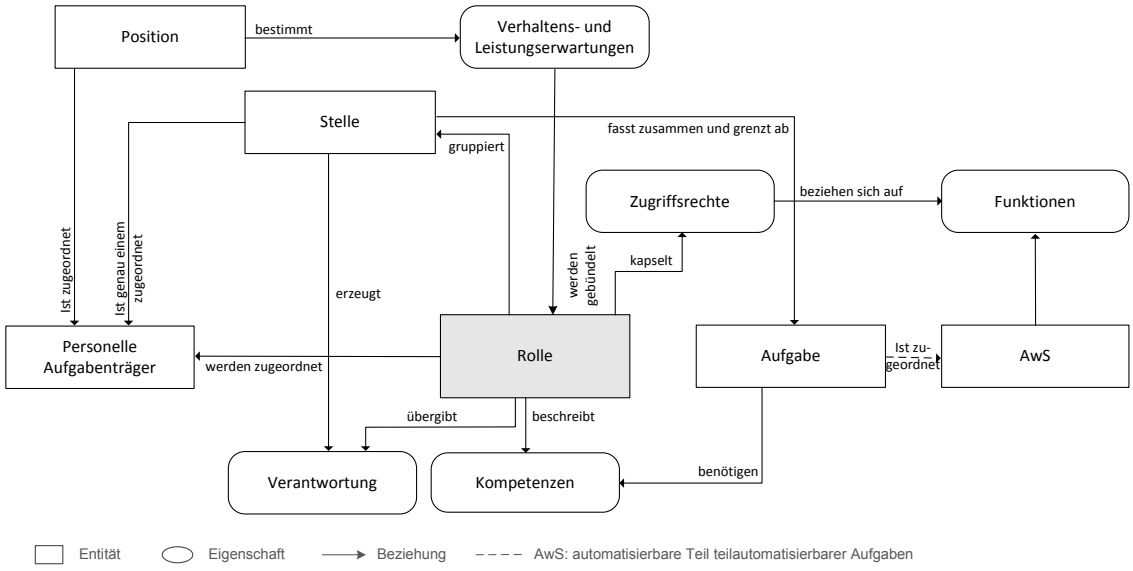

Abb. 5-7 Einordnung von AwS in das ganzheitliche Rollenkonzept

Das um das AwS erweiterte ganzheitliche Rollenkonzept zeigt **Abb. 5-7**. Der automatisierbare Teil einer teilautomatisierbaren Aufgabe wird maschinellen Aufgabenträgern - einem AwS - zugeordnet. Eine Anwendungssoftware kann in Funktionen zerlegt werden. Funktion in diesem Sinne sind Elemente von Vorgängen, die funktional beschreibbar sind (Ferstl und Sinz 2013, S. 61–64). Diese Funktionen bezeichnen fachliche Objekte mit den aufrufbaren Operatoren, z. B. Datenblatt.ausdrucken. Operatoren sind von extern zur Verfügung stehende Methoden, mit denen Datenobjekte aufgerufen und gegebenenfalls geändert werden können. Damit korrespondieren Objekte mit ihren Operatoren mit den Funktionen eines Anwendungssystems. Zugriffsrechte beziehen sich auf extern aufrufbare Funktionen, die in ihrer Gesamtheit das AwS darstellen. Rollen kapseln diese Zugriffsrechte und während der Laufzeit entscheidet die Rechteprüfung, ob der Zugriff erlaubt ist oder nicht. Vor Aufruf einer Funktion im AwS muss die Autorisierung durch eine Rechteprüfung anhand der Parameter Aufgabenträger, Objekt und Operator durchgeführt werden und entscheiden, ob dem Aufgabenträger das Recht auf Zugriff eingeräumt wird.

Ein AwS lässt sich nach dem sog. ADK-Strukturmodell gliedern. Kommunikation „wird weiter in die Bereiche Kommunikation mit Personen (K_P) und Kommunikation mit weiteren Maschinen (K_M) zerlegt“ (Ferstl und Sinz 2013, S. 321–323). Für die weitere Betrachtung in Bezug auf Rollen ist die Kommunikation K_P interessant. An dieser Schnittstelle muss entschieden werden, für welche Funktionen einem Aufgabenträger die Autorisierung erteilt werden kann. Die Rechteprüfung findet hier an der Kommunikationsschnittstelle zwischen personellen Aufgabenträgern und maschinellen Aufgabenträgern statt.

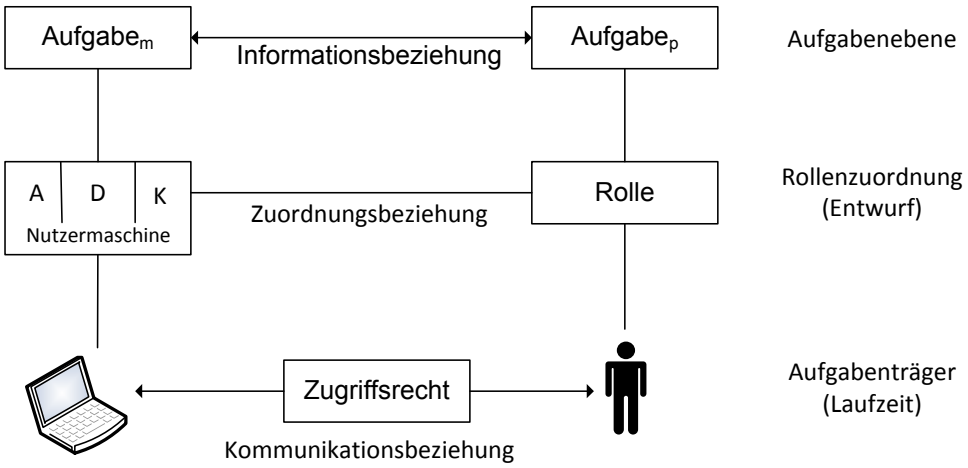

Abb. 5-8 Beziehung zwischen Aufgabenebene, Aufgabenträgerebene und Rolle

Bei der Zugriffskontrolle muss zwischen Design der Modellierung der Rollen und Zugriffsrechte, dem Speichern dieser Zugriffskontrollinformationen, der Rechteverwaltung, und der zur Laufzeit stattfindenden Rechteprüfung unterschieden werden. Auf Aufgabenebene besteht zwischen einer Aufgabe $\mathbf{A_m}$ und $\mathbf{A_p}$ eine Informationsbeziehung, die auf Aufgabenträgerebene zu einer Kommunikationsbeziehung führt. Während der Modellierung werden an Hand der Aufgaben, Aufbauorganisation und Ablauforganisation die Rollen festgelegt, die notwendig sind, um die Autorisierung durchführen zu können. Die Herleitung und Definition der Rollen setzt bei der Aufgabe und der Organisation an und berücksichtigt dabei, wie im vorherigen Kapitel im ganzheitlichen Rollenkonzept beschrieben, Stelle, Funktion, Position und Status im Unternehmen. Des Weiteren werden die Abteilungsmitgliedschaft sowie die notwendigen Qualifikationen des Aufgabenträgers einbezogen. Die Rollen übertragen damit Verantwortung und Kompetenz an Aufgabenträger. Während der Rechteprüfung (Laufzeit) werden die beim Entwurf modellierten Rollen ausgewertet. Es entsteht zur Laufzeit ein Zugriffsrecht, welches aussagt ob es einem Aufgabenträger erlaubt ist

das Objekt **x** mit dem Operator **o** aufzurufen. **Abb. 5-8** verdeutlicht diesen Zusammenhang.

Durch Zuordnung der Aufgaben zu Stellen entsteht die Aufbauorganisation. Durch die Aufgaben zusammen mit der Aufbauorganisation, den Funktionen und Kompetenzen können Rollen entwickelt und festgelegt werden (siehe Kapitel 5.1.2). Rollen werden Aufgabenträgern, die zur Erledigung ihrer Aufgaben ein AwS benötigen, zugeordnet. Die Rolle dient als Bindeglied zwischen Aufgabenträgern und Funktionen, die ein AwS zur Verfügung stellt. Sie bündelt die korrespondierenden Zugriffsrechte zu den möglichen Funktionen eines AwS. Ein Aufgabenträger erhält über die zugeordnete Rolle die Zugriffsrechte und die Erlaubnis, Funktionen aufrufen zu können, die er zur Erledigung einer Aufgabe benötigt.

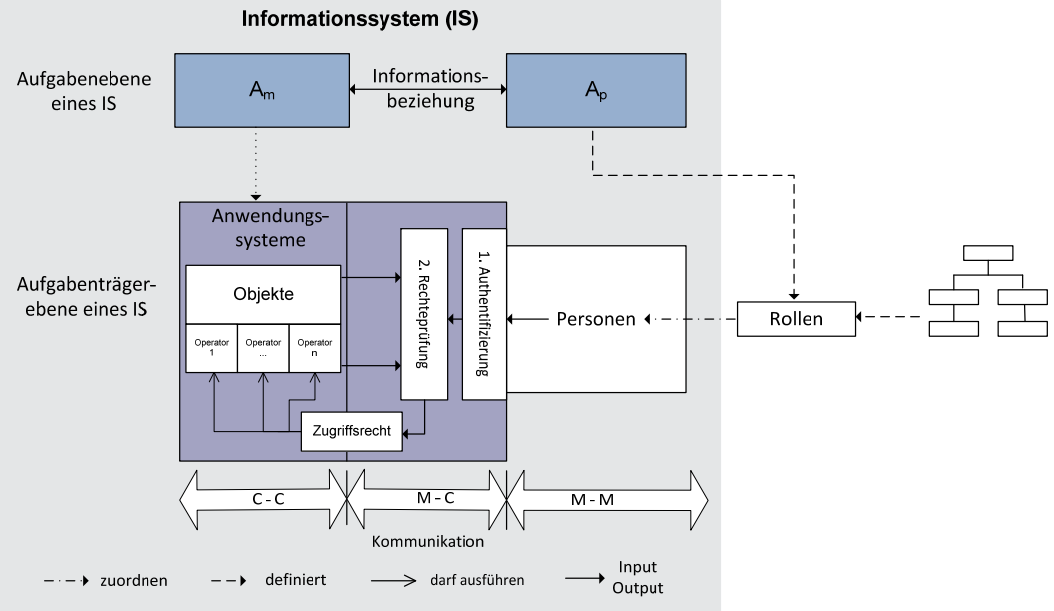

Abb. 5-9 Erweiterung der Informationsbeziehungen und Kommunikationssysteme im IS um Authentifizierung und Zugriffskontrolle

Die Einordnung von Rollen, Zugriffsrechten, Authentifizierung und Zugriffskontrolle in die Informationsbeziehungen und Kommunikationssysteme des IS stellt **Abb. 5-9** dar. Rollen werden anhand der Aufgaben und der Organisationsstruktur innerhalb des IS entwickelt, Personen zugeordnet und beides in der Rechteverwaltung gespeichert. An der Schnittstelle M-C muss zur Gewährleistung der IT-Sicherheit zuerst eine Authentifizierung und dann eine Rechteprüfung durchgeführt werden. Nach erfolgreicher Authentifizierung wird geprüft, ob der personelle Aufgabenträger über die Rolle das Zugriffsrecht besitzt, das mit der von ihm aufgerufenen

Funktion korrespondiert. Jeder Zugriff über die Rechteprüfung auf Funktionen des Anwendungssystems wird dabei protokolliert.

5.6 Rolle in der Unternehmensarchitektur

Nach der Einordnung der Rolle in die Informationsbeziehungen und Kommunikationssysteme des IS folgt im nächsten Schritt die Einordnung der Rolle in die Unternehmensarchitektur, um dabei zu analysieren, ob sich Rollen aus der Aufgabenebne heraus modellieren lassen. Zunächst wird die Unternehmensarchitektur nach dem Semantischen Objektmodell (SOM) (Ferstl und Sinz 2013, S. 195) dargestellt, um darin die Rolle einzuordnen. Anschließend wird ein Metamodell der Rollenzuordnung entwickelt. Die in Kapitel 5.4 herausgearbeiteten Eigenschaften einer Rolle als virtueller Aufgabenträger werden dabei aufgegriffen.

5.6.1 Einordnung der Rolle in die Unternehmensarchitektur

„Das umfassende Modellsystem eines betrieblichen Systems weist im Allgemeinen eine hohe Komplexität auf. Um diese Komplexität zu reduzieren, wird das Modellsystem in Teilmodellsysteme unterteilt“ (Ferstl und Sinz 2013, S. 195). Jedes Teilmodellsystem beschreibt das Objektsystem unter verschiedenen Blickwinkeln. Die Unternehmensarchitektur des Semantischen Objektmodells gliedert das Modellsystem in: Unternehmensplan, Modell der Geschäftsprozesse und das Ressourcenmodell (Ferstl und Sinz 2013, S. 194–197). Der linke Teil der **Abb. 5-10** zeigt diese Architektur.

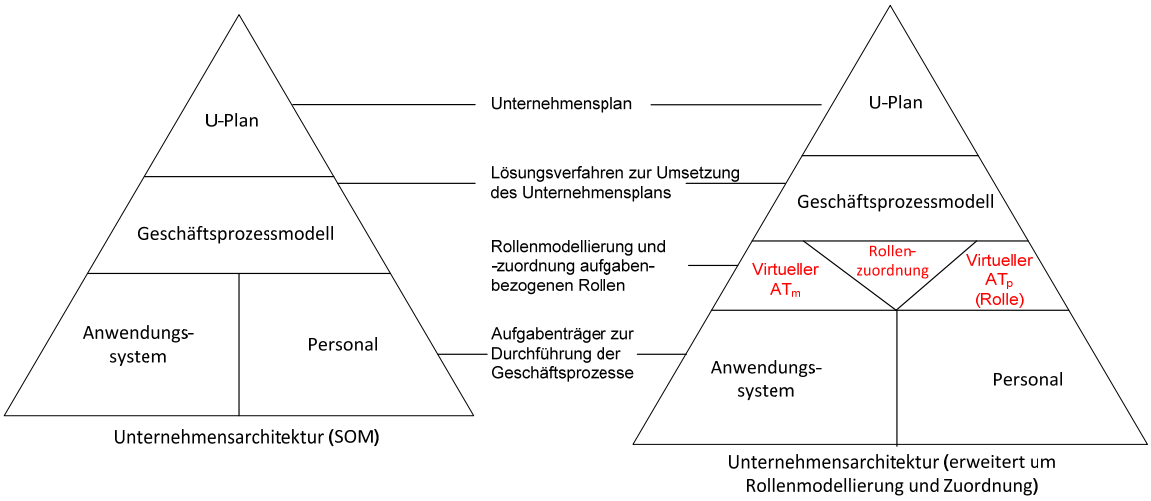

Abb. 5-10 Unternehmensarchitektur (SOM) (Ferstl und Sinz 2013, S. 195) und Erweiterung um die Rollenzuordnung

Die Außensicht als ein Teilmodellsystem eines betrieblichen Systems ist der Unternehmensplan. Das Geschäftsprozessmodell beschäftigt sich mit der Innensicht des betrieblichen Systems, der Aufgabenebene. „Es spezifiziert die Lösungsverfahren für die Realisierung des Unternehmensplans“ (Ferstl und Sinz 2013, S. 196). „Das Ressourcenmodell ist das Teilmodellsystem der Innenperspektive […] auf der Aufgabenträgerebene“ (Ferstl und Sinz 2013, S. 197). Ressourcen sind Personal, organisiert in Aufbauorganisationen, Anwendungssysteme, Maschinen und Anlagen. Die Spezifikation der personellen Aufgabenträger wird in Form der Aufbauorganisation modelliert. Anwendungssysteme sowie Maschinen oder Anlagen werden in Form der zugehörigen Spezifikationen modelliert (Ferstl und Sinz 2013, S. 197). Für die Autorisierung wird von der Spezifikation der Maschinen und Anlagen abstrahiert, da die Schnittstelle für die Mensch-Computer-Kommunikation zwischen personellen und maschinellen Aufgabenträgern in den Mittelpunkt gestellt wird.

Die Rechteprüfung muss an der Mensch-Computer-Schnittstelle durchgeführt werden, damit ein personeller Aufgabenträger seine Aufgaben erledigen kann. Die Rollenzuordnung verbindet die beiden Modellebenen **Geschäftsprozessmodell** und die **Spezifikation der Aufgabenträger**. Bei der Rollenzuordnung müssen die teilautomatisierten Aufgaben berücksichtigt werden. In **Abb. 5-10** wird die Unternehmensarchitektur nach SOM um die Perspektive der Rollenzuordnung erweitert.

Auf Ebene des Geschäftsprozessmodells wird die Aufgabenebene definiert. Auf Ebene der Rollenzuordnung werden aus den nichtautomatisierbaren Teilen teilautomatisierbarer Aufgaben die virtuellen personellen Aufgabenträger (Rollen) entwickelt. Die Rollen werden zu einer Rollenhierarchie verknüpft und bilden die Basis der Zugriffskontrolle. Parallel dazu werden für jede Aufgabe, die von außen zugreifbare Funktion eines AwS benötigt, die entsprechenden Zugriffsrechte erzeugt und Rollen zugeordnet. Für jede von außen zugreifbare konkrete Funktion wird ein Zugriffsrecht benötigt. Diese Funktionen werden zu virtuellen maschinellen Aufgabenträgern zusammengefasst. Danach wird der automatisierbare Teil einer teilautomatisierbaren Aufgabe virtuellen maschinellen Aufgabenträger zugeordnet. Dadurch wird deutlich, dass sich Rollen aus den Aufgabenebene im Top-Down-Ansatz modellieren lassen.

Aus den Rollenkonzepten, der Einordnung der Rollen in die Informationsbeziehungen und Kommunikationssysteme des IS und die Unternehmensarchitektur wird nun im nächsten Schritt ein Metamodell der Rollenzuordnung entwickelt.

5.6.2 Metamodell der Rollenzuordnung

„Ein Metamodell definiert die verfügbaren Arten von Modellbausteinen, die Arten der Beziehungen zwischen den Modellbausteinen, die Regeln für die Verknüpfung von Modellbausteinen durch Beziehungen sowie die Bedeutung (Semantik) der Modellbausteine und Beziehungen“ (Ferstl und Sinz 2013, S. 137). Mit dem Metamodell der Rollenzuordnung soll ein Modell beschrieben werden, das das Modell der Zuordnung von Rollen zu personellen Aufgabenträgern aufzeigt. Als Grundlage für dieses Metamodell wird Bezug genommen auf die Einführung des Meta-Metamodell in Bild 5.5 von (Ferstl und Sinz 2013, S. 139) und dem Metamodell und Beschreibung für Geschäftsprozessmodelle von (Ferstl und Sinz 2013, S. 210) und (Krumbiegel 1997, S. 129–140).

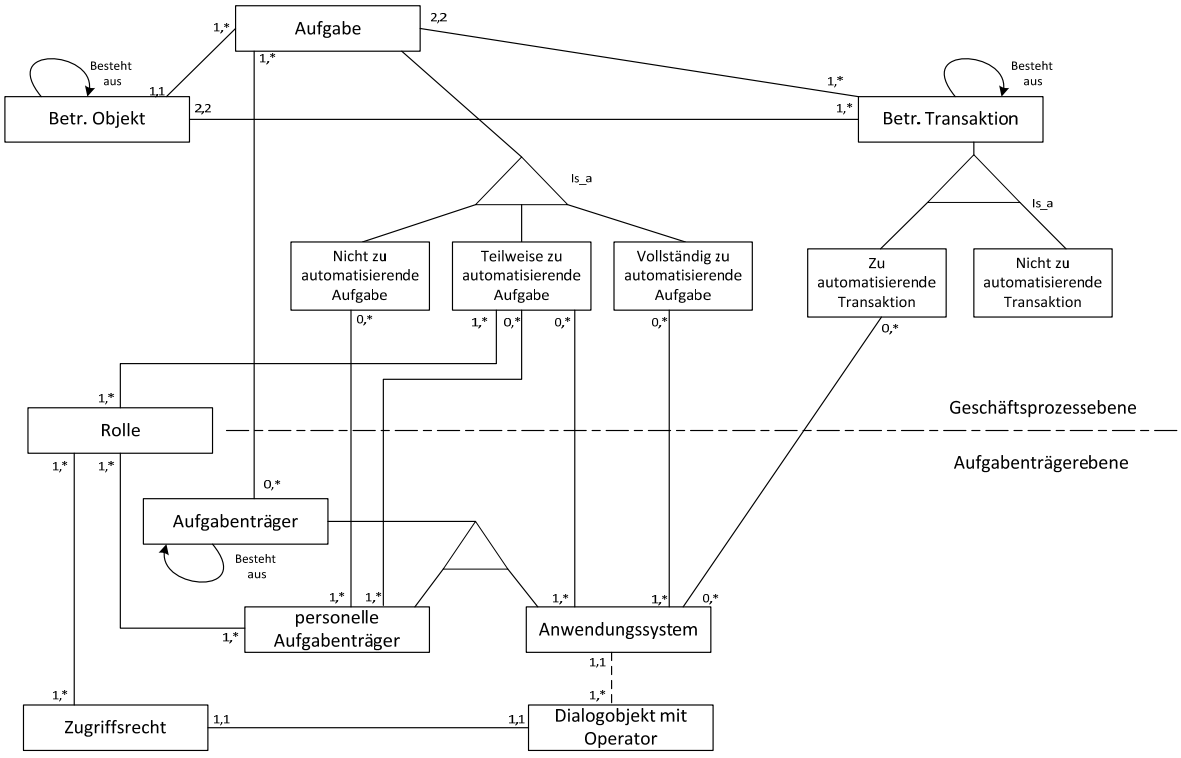

Abb. 5-11 Metamodell der Rolleneinordnung

Das Metamodell der Rollenzuordnung zeigt **Abb. 5-11**. Ein betriebliches Objekt ist mit einer oder beliebig vielen Transaktionen verbunden. Umgekehrt verbindet jede Transaktion genau zwei Objekte. Eine Transaktion wird durch zwei Aufgaben

durchgeführt. Die Aufgaben sind vollständig und disjunkt spezialisiert zu nicht, teilweise oder vollständig automatisierten Aufgaben. Bei der Zuordnung zu Aufgabenträgern werden Aufgaben bezüglich der Automatisierungsanforderung in nicht, teilweise oder vollständig automatisierbar differenziert (Ferstl und Sinz 2013, S. 220; Krumbiegel 1997, S. 130–131). Die Unterscheidung in zu automatisierende und nicht zu automatisierende Transaktionen ist das Ergebnis der vollständigen und disjunkten Spezialisierung von betrieblichen Transaktionen.

Aufgabenträger werden ebenso disjunkt eingeteilt in Anwendungssysteme und personelle Aufgabenträger. Jede Aufgabe wird mindestens einem bis beliebig vielen Aufgabenträgern zugeordnet. Jedem Aufgabenträger wird mindestens eine Aufgabe zugeordnet. Teilautomatisierte Aufgaben werden gemeinsam von personellen und maschinellen Aufgabenträgern durchgeführt. Automatisierte Aufgaben werden ausschließlich von maschinellen Aufgabenträgern und nicht automatisierte Aufgaben von personellen Aufgabenträgern ausgeführt.

Jede teilweise automatisierte Aufgabe kann von einer bis beliebig vielen Rollen zugeordnet werden. Umgekehrt kann eine Rolle für eine bis beliebig viele Aufgaben zuständig sein. Die Beziehungskante zwischen Aufgaben und Rollen gewährleistet, dass jede Aufgabe zumindest zu einer modellierten Rolle gehört. Eine Rolle muss mindestens einem bis zu beliebig vielen personellen Aufgabenträger zugeordnet werden. Einem personellen Aufgabenträger muss mindestens eine Rolle zugeordnet sein, er kann aber beliebig viele Rollen erhalten. Eine Rolle bündelt dabei ein bis beliebig viele Zugriffsrechte und ein Zugriffsrecht muss mindestens einer Rolle, kann aber auch beliebig vielen Rollen zugeordnet werden.

Ein Anwendungssystem besteht aus mindestens einem Dialogobjekt mit Operator, also einer Funktion. In der Praxis besteht dieses aus beliebig vielen Funktionen. Eine Funktion ist genau einem Anwendungssystem zugeordnet. Ein Zugriffsrecht bezieht sich immer auf genau eine von außen zugreifbare Funktion innerhalb eines Anwendungssystems.

5.7 Zusammenfassung

Der Begriff Rolle wird unterschiedlich, aber nicht immer disjunkt, in den einzelnen Forschungsbereichen verwendet. Durch die systematische Analyse des Begriffs Rolle und Einordnung in die verschiedenen Rollenkonzepte wurden die Gemeinsamkeiten

herausgearbeitet und daraus ein ganzheitliches Rollenkonzept entwickelt. In allen untersuchten Rollenkonzepten wird die Rolle als **Aufgabenträger** verwendet. Die Rolle besitzt alle Eigenschaften der Aufgabenträger, ist aber nicht auf die Kapazität eines Aufgabenträgers beschränkt. In allen Rollenkonzepten stellt sie eine Repräsentation des konkreten Aufgabenträgers dar und kann als **virtueller Aufgabenträger** bezeichnet werden.

Durch das Einbinden der Rolle in das betriebliche Informationssystem und das ADK-Modell wird gezeigt, dass sie sich als virtueller Aufgabenträger wie eine Zwischenschicht zur Zuordnung von Aufgaben zu Aufgabenträgern einführen lässt. Die Rolle ist damit an der Schnittstelle zwischen dem Geschäftsprozessmodell und der Aufgabenträgerzuordnung anzusiedeln. Die Ergebnisse der Untersuchungen fließen in die Erweiterung der Unternehmensarchitektur um die Rollenzuordnung ein und zeigen, dass Rollen als virtuelle Aufgabenträger Top-Down aus der Aufgabenebene des IS heraus modellierbar sind. Das entwickelte Metamodell der Rollenzuordnung zeigt, dass die Rolle Zugriffsrechte, dem Gegenstück der von außen aufrufbaren Funktionen des AwS, die personellen Aufgabenträgern zugeordnet werden, bündelt. Damit stehen dem personellen Aufgabenträger die notwendigen Zugriffsrechte zur Verfügung, um die ihm über die Stellen zugeordneten Aufgaben im IS erfüllen zu können.

6 Erweitertes rollenbasiertes Zugriffskontrollmodell - eRBAC

Im vorherigen Kapitel konnte gezeigt werden, dass das Konzept der Rolle in das Konzept des betrieblichen Informationssystems integriert werden kann. Damit wurde nochmals die in Kapitel 3 getroffene Auswahl des Zugriffskontrollmodells RBAC untermauert. Das Zugriffskontrollmodell RBAC und die untersuchten Konzepte in Kapitel 4 bilden eine Grundlage[53] für die Entwicklung des erweiterten rollenbasierten Zugriffskontrollmodells eRBAC. Für eRBAC wurden folgende Konzepte aus Kapitel 4 modifiziert übernommen: Die Rollenhierarchie wird um virtuelle Rollen erweitert und als Zugriffsrechtsvererbung interpretiert. Die Rechteverwaltung wird durch sog. Administrationsrollen geregelt. Das Konzept der Delegation wurde auf Basis von PBDM1 eingeführt. Außerdem wird die Domänenbeschränkung für Rollen ermöglicht. Neu aufgenommen wurde das Konzept der Objekttypen, um zwischen dem Aufruf einer Anwendung und der Zugriffskontrolle innerhalb eines Anwendungssystems unterscheiden zu können. Darüber hinaus wurde eine Personalisierung der Rolle eingeführt, um Informationen aus dem Anwendungssystem nach der Authentifizierung zur Verfügung zu haben. Zusammen mit dem Konzept der Objekttypen lässt sich damit eine flexible, benutzerfreundliche Aufrufstruktur zwischen Authentifizierung und Zugriffskontrolle realisieren. Nach der Beschreibung von eRBAC wird dieses grafisch dargestellt und damit gezeigt, dass es das Referenzmodel RBAC formal erweitert.

Zunächst werden die Entitäten des erweiterten rollenbasierten Zugriffskontrollmodells eRBAC, aufbauend auf den Kapiteln 2.4.2.2, 3 und 4 vorgestellt. Die in Kapitel 3.1.3 festgelegten Anforderungen an Zugriffskontrollmodelle werden dabei berücksichtigt.

6.1 Objekttypen, Objekte, Operatoren und Zugriffsrechte

Neben den beschriebenen Entitäten Objekt, Operator und Zugriffsrecht (Kapitel 2.4.2.2) wurde zur Typisierung der Objekte die zusätzliche Entität **Objekttyp** eingeführt, um damit eine Strukturierung der Objekte zu erreichen.

[53] Es wird hier auf in Kapitel 4 beschriebenen Grundlagen dieser Konzepte verwiesen.

6.1.1 Objekttypen

In Anlehnung an die Einteilung in strukturelle und funktionale Rollen (siehe Kapitel 4.8) (Coyne und Davis 2008, S. 51–52) wird in eRBAC zwischen dem Öffnen einer Anwendungssoftware und dem Aufruf von Objekten innerhalb einer Anwendungssoftware unterschieden. Diese Unterscheidung wird nicht anhand von Rollen getroffen, sondern durch eine Typisierung der Objekte. Es werden folgende zwei Objekttypen definiert:

- **Anwendung** für Objekte, die den Einstiegspunkt einer Anwendungssoftware darstellen.
- **Klasse** für Objekte, die Klassen innerhalb einer Anwendungssoftware repräsentieren.

Zu jedem Objekttyp existieren beliebig viele Objekte. Jeder Rolle sind mindestens **ein** Objekt des Typs **Anwendung** und beliebige viele Objekte des Typs **Klasse** zugeordnet. Ein zugeordnetes Subjekt muss die Möglichkeit besitzen, eine Anwendung zu öffnen, um dann Funktionen innerhalb der Anwendungssoftware aufrufen zu können. Durch die Einteilung in Objekttypen wird im Gegensatz zur Einteilung in strukturelle und funktionale Rollen die Anzahl der Rollen nicht unnötig vergrößert.

Ruft das Subjekt eine Anwendung auf, wird damit auch implizit die dazugehörige Rolle mit ihren Zugriffsrechten aktiviert. Dadurch ist zu einem Zeitpunkt immer genau eine Rolle aktiv und wird das Prinzip der minimalen Zugriffsrechte (siehe Kapitel 2.4.3) realisiert.

Mit der Einteilung der Objekte in die Objekttypen **Anwendung** und **Klasse** kann eine flexible benutzerfreundliche Aufrufstruktur durch eine, wenn möglich automatische Weiterleitung zwischen Authentifizierung und Zugriffskontrolle, implementiert werden. Dazu besitzt ein Objekt des Typs **Anwendung** zwei zusätzliche Attribute: **Aufrufadresse** und **Aufrufbezeichnung**. Das Attribut Aufrufadresse speichert die Adresse der aufzurufenden Anwendung und das Attribut Aufrufbezeichnung enthält den Namen der Anwendung zur Darstellung im Menü.

6.1.2 Objekte, Operatoren und Zugriffsrechte

Objekte und Operatoren können in eRBAC unabhängig voneinander definiert werden, so dass theoretisch jede beliebige Kombination möglich ist. Durch die explizite

Zuordnung von Operatoren zu Objekten werden die erlaubten Zugriffsrechte festgelegt. Für die Administration wird damit ausgeschlossen, dass Rollen aus Gründen der Informationssicherheit nicht wünschenswerte Kombinationen von Objekten und Operatoren als Zugriffsrechte zugeordnet werden. Die Zugriffsrechte können objektspezifisch festgelegt werden.

In **Tab. 6-1** wird dieses Vorgehen beispielhaft für die Domäne eines Prüfungsverwaltungssystems dargestellt. In den Spalten sind ausgewählte Objekte, in den Zeilen ausgewählte Operatoren aufgelistet. An dem Schnittpunkt einer Spalte und Zeile ergibt sich das resultierende Zugriffsrecht. Ein Objekt des Typs Anwendung besitzt immer nur den Operator **open**. Einem Objekt des Typs Klasse können beliebige objektspezifische Operatoren zugeordnet werden.

Tab. 6-1 Objekt, Operatoren und zugelassene Zugriffsrechte

Objekte Operatoren	Session	Fak	Datenblatt	Prfstd	Lehrstuhl-modul
Open					Lehrstuhl-modul. Open
Read	Session. read		Datenblatt. read		
readPDF			Datenblatt. readPDF		
Update	Session. update				
Insert	Session. insert				
storeCSV				Prfstd.storeCSV	
getByID		Fak.getByID			

Die Objekte können anwendungsspezifisch hinterlegt werden. Die Granularität des zu überprüfenden Objektes mit seinem Operator kann sehr unterschiedlich sein. Es kann sich auf das Lesen eines Tupels in einer einzelnen Tabelle beziehen (*Fak.getByID*) oder auf ein komplexes fachliches Objekt, das über viele Tabellen hinweg eine Aggregation und Projektion von Daten (*Datenblatt.read*) darstellt.

Das Konzept der negativen Zugriffsrechte (siehe Kapitel 4.6) wurde aus folgenden Gründen nicht in eRBAC übernommen: Es wird in eRBAC die höhere Sicherheit des geschlossenen Systems präferiert. Die Delegation von Zugriffsrechten ist in eRBAC möglich, aber es können nur Zugriffsrechte delegiert werden, die von der Administration als delegierbar gekennzeichnet sind und die der Delegierende selbst besitzt. Eine regelbasierte Subjektzuordnung zur Laufzeit ist in eRBAC nicht vorgesehen.

6.2 Rollentypen und Rollen in eRBAC

Die Modellierung und Konzeption von Rollen stellen den wichtigsten Teil bei der Umsetzung eines Zugriffskontrollsystems dar. Das Modell eRBAC unterteilt die Rollen in Rollentypen, so dass getrennte Sicherheitsrichtlinien für den jeweiligen Rollentyp hinterlegt werden können.

Zu jedem Rollentyp existieren beliebig viele Rollen. Zur Definition der einzelnen Typen werden hierzu drei Konzepte für eRBAC übernommen bzw. angepasst:

1. Das Anlegen von virtuellen Rollen, um gemeinsame Zugriffsrechte zu kapseln (siehe Kapitel 4.3.1).
2. Das Zugriffskontrollmodell RBAC wird für eine dezentrale Administration der Rechteverwaltung verwendet (siehe Kapitel 4.4).
3. Die Möglichkeit der Delegation wird zugelassen (siehe Kapitel 4.5).

Die folgenden Rollentypen sind in eRBAC definiert:

- Anwendungsrollen[54]
- Virtuelle Rollen
- Administrationsrollen
- Delegationsrollen

Anwendungsrollen beziehen sich auf die Ebene der Anwendungssoftware. Diese Rollen bündeln die Zugriffsrechte vom Öffnen der Anwendung bis hin zur Überprüfung jedes einzelnen Zugriffs auf von außen zugreifbare Funktionen eines Anwendungssystems. Rollen dieses Typs können in die Rollenhierarchie eingebunden werden, aber auch direkt einem Subjekt zugeordnet werden.

Virtuelle Rollen sind spezielle Anwendungsrollen und kapseln Zugriffsrechte, die von mehreren Rollen benötigt werden. Diese werden innerhalb der Rollenhierarchie eingeordnet, womit eine größere Übersichtlichkeit über die Zugriffsrechte einer Rolle entsteht. Diese virtuellen Rollen dürfen nur in der Rollenhierarchie verwendet und nicht direkt einem Subjekt zugeordnet werden.

Administrationsrollen bündeln alle Zugriffsrechte, die für die Rechteverwaltung erforderlich sind. Es können beliebig viele Administrationsrollen angelegt werden. Diese sind jedoch disjunkt von den Rollen der anderen Rollentypen.

[54] Im weiteren nur Rolle genannt

Delegationsrollen können benutzt werden, um einen Teil oder alle Zugriffsrechte einer Rolle zu delegieren. Delegationsrollen sind disjunkt von Anwendungsrollen und Administrationsrollen. Die delegierbaren Zugriffsrechte bündeln eine Teilmenge der Zugriffsrechte einer Anwendungsrolle, die von der Administration als delegierbar gekennzeichnet wurden.

6.3 Statische und dynamische Aufgabentrennung

In eRBAC wird die Aufgabentrennung durch sich ausschließende Rollen umgesetzt. Die statische Aufgabentrennung wird dadurch gewährleistet, indem es bei der Administration nicht möglich ist, einem Subjekt zwei sich ausschließende Rollen, auch innerhalb der Rollenhierarchie, zuzuordnen. Die dynamische Aufgabentrennung muss durch das Zugriffskontrollsystem sichergestellt werden, da es nicht möglich sein darf, zwei sich ausschließende Rollen gleichzeitig zu aktivieren. Sowohl der funktionale Aspekt, als auch das 4-Augenprinzip kann je nach organisatorischer Anforderung in eRBAC mit statischer oder dynamischer Aufgabentrennung umgesetzt werden.

Der Vorschlag im Referenzmodell (siehe Kapitel 3.2.7.4) die Aufgabentrennung durch die Angabe einer Menge an möglichen Rollen und einer Kardinalität zu konzipieren, wird in eRBAC nicht weiter verfolgt. Dieses Konzept ist nicht geeignet, eine Aufgabentrennung nach fachlichen Gesichtspunkten zu realisieren, da im Referenzmodell nur die Anzahl der zugeordneten bzw. aktivierten Rollen darüber entscheidet, ob eine weitere Rolle zugeordnet oder aktiviert werden darf.

6.4 Rollenhierarchie

Die Rollenhierarchie in eRBAC wird anhand einer allgemeinen Rollenhierarchie (Kapitel 3.2.7.3) gebildet. Sie lässt eine Mehrfachvererbung zu und bildet ausschließlich die Zugriffsrechtsvererbung, den intensionalen Aspekt (Kapitel 4.3.2) einer Rollenhierarchie, ab. Durch die Mehrfachvererbung innerhalb einer Rollenhierarchie ist es zulässig, dass eine Rolle von mehr als einer Rolle erbt. Damit es zu keinem Konflikt zwischen Aufgabentrennung und Rollenhierarchie kommt, muss zuerst die Aufgabentrennung definiert werden, bevor eine Rolle in die Rollenhierarchie eingebunden werden darf. Dabei stellt die Implementierung der Rechteverwaltung sicher, dass im Pfad der Hierarchie ein Subjekt nicht von zwei sich ausschließenden

Rollen erbt. Für die Umsetzung der dynamischen Aufgabentrennung wird eine zweite getrennte Rollenhierarchie als Aktivierungshierarchie modelliert. Diese dient dazu die Aufgabentrennung bei der Aktivierung der Rollen berücksichtigt.

6.5 Administration in eRBAC

Die Administration in eRBAC erfolgt über Administrationsrollen. Bei der Konzeption dieser wird von den grundlegenden Definitionen aus Kapitel 4.4 ausgegangen. Die Rollen und Zugriffsrechte für die Administration sind disjunkt von den Rollen der anderen Rollentypen. Die beiden Basiskomponenten Vorbedingung und Rollenbereiche orientieren sich nicht an der bestehenden Subjekt- bzw. Zugriffsrechtszuordnung, sondern an davon separat definierten Subjekt- bzw. Zugriffsrechtepools, die sich am Aufbau der Organisation orientieren.

Die den Administrationsrollen zugeordneten Subjekte können folgende Aufgaben wahrnehmen:

- **Grundlagen**: Objekte, Operatoren und Zugriffsrechte neu anlegen.
- **Rollenverwaltung**: Rollen anlegen, Zugriffsrechte zuordnen, Aufgabentrennung festlegen und Rollen in eine Rollenhierarchie einordnen.
- **Subjektverwaltung**: Subjekte mit ihren Zugangsdaten anlegen und die Subjektzuordnung vornehmen.
- **Delegationsverwaltung**: Festlegen, welche Zugriffsrechte und Rollen delegierbar sind.
- **Domänenverwaltung**: Zuordnen von Schlüsselwerten aus dem Anwendungssystem zur Festlegung der Attribute für die Personalisierung und die Domänenbeschränkung.

6.6 Administrationsbedingte Delegation in eRBAC

Delegation ist ein Mittel, Zugriffsrechte über Rollen auf Zeit an ein anderes Subjekt zu übertragen. Folgende auf PBDM1 basierende Konzepte (siehe Kapitel 4.7) werden in eRBAC übernommen:

- Dauer: Die Subjektzuordnung kann mit einer Zeitangabe versehen werden, falls die Delegation nur zeitlich beschränkt werden soll.
- Monotonie: Es wird das Konzept der monotonen Delegation umgesetzt. Der Delegierende behält seine Zugriffsrechte.

- Gesamtheit: Es müssen die Zugriffsrechte der zu delegierenden Rolle nicht in ihrer Gesamtheit delegiert werden.
- Administration: Die Delegation wird vom Besitzer der Rolle vorgenommen.
- Delegationsstufen: Es wird eine einstufige Delegation unterstützt.
- Mehrfache Delegation: Es kann eine Rolle vom Delegierenden an mehr als ein Subjekt delegiert werden oder ein Subjekt kann für mehr als einen Delegierenden Delegierter sein.
- Vereinbarung: Der Delegierende entscheidet alleine über die Delegation.

Das Delegationskonzept für eRBAC sieht eine von der Administration überwachte Delegation vor. Die Administration legt fest, ob eine Rolle delegiert werden darf. Damit kann verhindert werden, dass ohne Wissen der Administration umfangreiche Rollen delegiert werden. In einer zweiten Stufe wird bei der Zugriffsrechtszuordnung hinterlegt, welche Zugriffsrechte delegierbar sind. Innerhalb der Delegationsrollen existieren keine Hierarchien. Wird eine Delegationsrolle erzeugt, kann diese alle zugeordneten als delegierbar gekennzeichneten Zugriffsrechte aus dem Pfad der Hierarchie einschließlich der virtuellen Rolle erhalten.

Wird von einem Subjekt eine Delegation vorgenommen, so erstellt der Delegierende eine neue Delegationsrolle (DRo) und ordnet dieser die gewünschten Zugriffsrechte zu. Die Delegationsrolle kann beliebigen dem Zugriffskontrollsystem bekannten Subjekten zugeordnet werden.

Es wird nur eine einstufige Delegation unterstützt, d. h. es ist einem Delegierten nicht möglich, eine Delegationsrolle weiter zu delegieren. Dabei ist dem Delegierenden eine Mehrfachdelegation erlaubt, d. h. eine Delegationsrolle kann mehr als einem Subjekt zugeordnet werden.

Neben der Festlegung der Delegation muss auch die Rücknahme einer Delegation definiert werden. eRBAC erlaubt eine Rücknahme der Delegation durch die Administration und dem Delegierenden selbst. Wird dem Delegierenden die Rolle entzogen, auf die sich die Delegationsrolle bezieht, werden auch allen Subjekten diese zugeordnete Delegationsrolle entzogen.

6.7 Personalisierung von Rollen

Durch eine Single-Sign-On-Authentifizierung wird zwar die Identität eines Subjektes unternehmensweit festgelegt, aber es sind nicht immer alle Attribute dieses Subjektes aus den eingesetzten Anwendungssystemen bekannt. Eine Personalisierung ist dann erforderlich, wenn nach der Authentifizierung Informationen, die sich auf Entitäten aus dem Zielanwendungssystem beziehen, benötigt werden. Die dazu notwendigen Daten müssen im Zugriffskontrollsystem hinterlegt werden, um die Daten an das Anwendungssystem übergeben und Informationen entsprechend filtern zu können. In eRBAC wird dieser Sachverhalt durch das Konzept der Personalisierung abgebildet.

Personalisierung der Rolle anhand von Entitäten aus dem Zielanwendungssystem

Eine Personalisierung in eRBAC wird anhand von Entitäten aus dem Zielanwendungssystem vorgenommen. Dafür wird bei der Rolle hinterlegt, welches Datenobjekt aus dem Zielanwendungssystem für die Personalisierung eines Subjektes herangezogen wird. Der Eintrag bezieht sich auf eine Entität und ist der Name der Tabelle aus der relationalen Datenbank des Zielanwendungssystems. Für jedes Subjekt werden nach der Subjektzuordnung (SZ) die Werte der Schlüsselattribute aus dem Zielanwendungssystem hinterlegt. Sie identifizieren ein Tupel in der Tabelle eindeutig. Konkret sind die Schlüsselattribute der Primärschlüssel des entsprechenden Datenobjektes.

In FlexNow werden anhand der Organisationseinheit die zu bearbeitenden Lehrveranstaltungen und Prüfungen eines Prüfenden festgelegt. Es reicht in FN2RBAC der Schlüssel der Organisationseinheit aus, um diesen an das Anwendungssystem FlexNow zu übergeben und damit die entsprechenden Daten filtern und anzeigen zu können. Die anzuzeigenden Daten werden von der Geschäftslogik ermittelt. Ein mögliches Nutzungsszenario an Hochschulen ist, dass Mitarbeiter für mehrere Lehrstühle arbeiten. Durch dieses Konzept kann ein Mitarbeiter für verschiedene Lehrstühle Notenlisten bearbeiten, ohne dafür unterschiedliche Zugangsdaten zu benötigen.

6.8 Domänenbeschränkung durch parametrisierte Rollen

Neben der Personalisierung ist auch eine Domänenbeschränkung erforderlich, da alle Subjekte einer zugeordneten Rolle dieselben Zugriffsrechte besitzen und damit dieselben zugehörigen Funktionen ausführen dürfen. Aber jedes Subjekt darf nur die erlaubten Daten sehen. Um nicht für jedes Subjekt eine persönliche Rolle anlegen zu müssen, wird die Domäne der Ergebnismenge über Parameter beschränkt. Dazu werden die Rollen parametrisiert.

> *Ein Prüfungsausschussvorsitzender kann Prüfungsdaten von Studierenden nur für Studienfächer seiner Fakultät abrufen. Ein weiteres Anwendungsbeispiel sind dezentrale, den einzelnen Fakultäten zugeordneten Prüfungsämter, da diese nur die Daten der Studierenden ihrer Fakultät bearbeiten dürfen.*

Diese domänenabhängigen Informationen werden durch parametrisierbare Rollen berücksichtigt werden. Mit parametrisierten Rollen wird ein höherer Grad an Genauigkeit und Flexibilität erreicht, ohne eine Vielzahl von Rollen modellieren zu müssen (Adams 2006, S. 1). Dazu werden in eRBAC Parameter hinterlegt, die je nach Zielanwendungssystem unterschiedlich sind. Diese Parameter werden den Rollen zugeordnet, für die eine Parametrisierung erforderlich ist. Nach der Subjektzuordnung werden jedem Subjekt die erlaubten Attributwerte für die Domänenbeschränkung eingetragen.

Die Domänenbeschränkung wird nach dem Erlaubnisprinzip (siehe Kapitel 2.4.3) durchgeführt. Es muss mindestens **ein** Attribut für einen Parametereintrag zugeordnet sein, ansonsten werden keine Daten ausgeliefert. Durch die Umsetzung der Domänenbeschränkung nach dem Erlaubnisprinzip wird zur besseren IT-Sicherheit ein zusätzlicher Aufwand bei der Rechteverwaltung akzeptiert. Ebenso kann sich der Administrationsaufwand durch die Datenmenge der einzuschränkenden Domäne erhöhen.

> *Es besteht die Möglichkeit, die Studiengänge der Studierenden, für die ein Prüfungsausschussvorsitzender zuständig ist, entweder auf Ebene der Fakultät oder auf Ebene der Studiengänge einzuschränken. Die Mächtigkeit der Menge der Domäne Fakultät umfasst je nach Hochschule 5 bis 20 Fakultäten, die der Studiengänge umfasst hingegen 100 bis 400 gleichzeitig zu verwaltende Studiengänge.*

6.9 Grafische Darstellung und Formalisierung von eRBAC

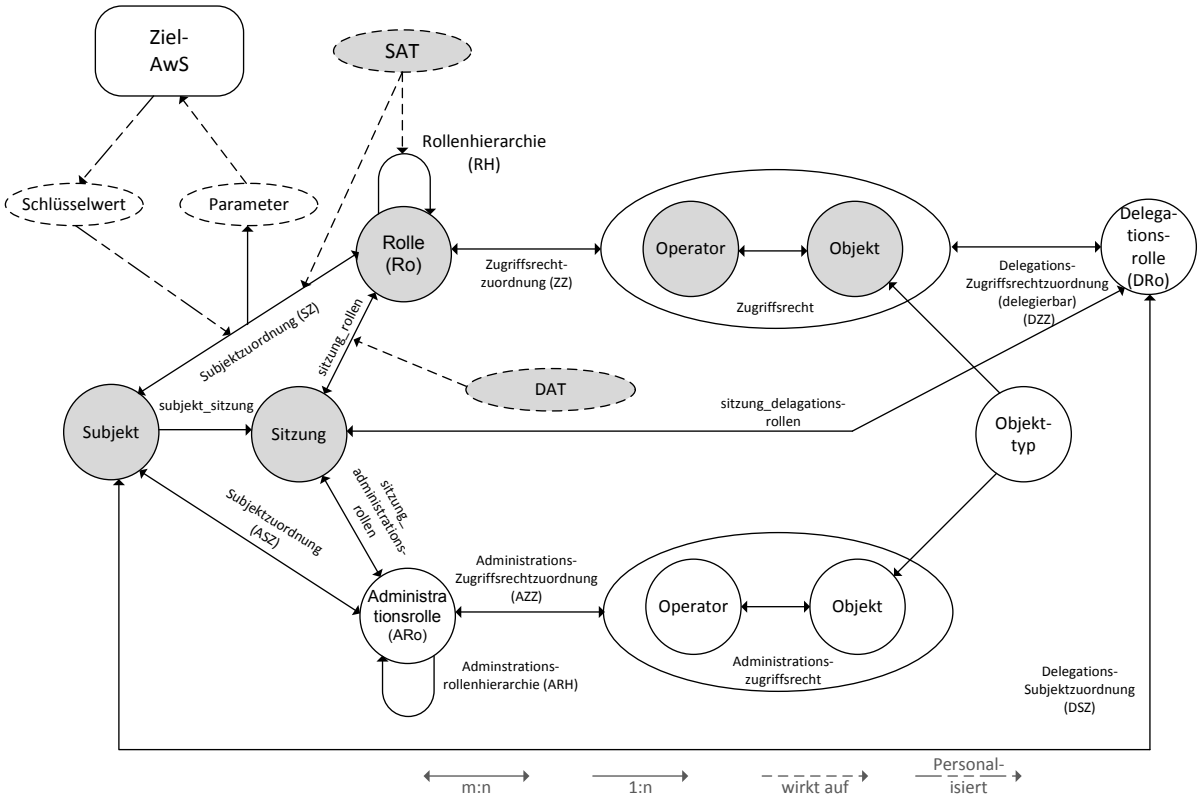

Abb. 6-1 Darstellung der Zusammenhänge der einzelnen Komponenten von eRBAC

In **Abb. 6-1** werden grafisch die Zusammenhänge von eRBAC dargestellt. Grundlage der Weiterentwicklung ist das Referenzmodell RBAC (siehe Kapitel 3.2.7)[55]. In dieser Grafik werden die Rollen der drei Rollentypen: Anwendungsrollen (im folgenden Rolle genannt), Administrationsrollen und Delegationsrollen dargestellt. Virtuelle Rollen werden wie Anwendungsrollen behandelt und werden in der Grafik nicht explizit dargestellt.

In der folgenden Definition wird das formale Modell von eRBAC vorgestellt. Damit kann gezeigt werden, dass die neuen Konzepte von eRBAC das formale Modell des Referenzmodells erweitern:

Definition 6-1 Formale Zusammenfassung von eRBAC[56] siehe **Abb. 6-1**

Kernmodell

– Subjekt (S), *Rollentyp (RT[57])*, Rolle (Ro), *Objekttyp (OT)*, Objekt (O) und Operator (Op), Sitzung (Si), *Parameter (P)*, *Schlüsselwert (Sw)*.

[55] Die bereits im Referenzmodell definierten Entitäten sind grau markiert.
[56] Die Definitionen wurden ANSI INCITS 359-2004 (2004) und Sandhu et al. (1999) entnommen und um die neuen Elemente von eRBAC erweitert. Die Erweiterungen sind kursiv gekennzeichnet.

- *OTO $\subseteq$ OT $\times$ O, eine 1:n Zuordnungsrelation von einem Objekttyp zu Objekten. Ein Tupel (oT, o) $\in$ OTO legt ein Objekt o $\in$ O zu einem Objekttyp oT $\in$ OT fest.*
- *RTRo $\subseteq$ RT $\times$ Ro, eine 1:n Zuordnungsrelation von einem Rollentyp zu Rollen. Ein Tupel (rT, Ro) $\in$ RTRo legt eine Rolle ro $\in$ Ro auf einem Rollentyp rT $\in$ RT fest. Dadurch werden Rollen als Anwendungsrollen (Ro), Administrationsrollen (ARo), virtuelle Rollen (vRo) und Delegationsrollen (DRo) festgelegt.*
- Subjektzuordnung SZ $\subseteq$ S $\times$ Ro, eine m:n Zuordnungsrelation von Subjekt zu Rollen. Ein Tupel (s, ro) $\in$ SZ legt die Rollenmitgliedschaft eines Subjektes s $\in$ S in einer Rolle ro $\in$ Ro und den damit verbundenen Zugriffsrechten fest.
- Die Funktion zugeordnete_subjekte: (ro:Ro) $\rightarrow$ 2^S mit zugeordnete_subjekte (r) = {s $\in$ S| (s, ro) $\in$ SZ} bestimmt die Menge der einer Rolle zugeordneten Subjekte.
- $Z = 2^{O} \times {}^{Op}$, eine Menge von Zugriffsrechten.
- Zugriffsrechtszuordnung ZZ $\subseteq$ Z $\times$ Ro, eine m:n Zuordnungsrelation von Zugriffsrechten zu Rollen. Über ein Tupel (z, ro) wird mit einer Rolle ro $\in$ RO ein benötigtes Zugriffsrecht z $\in$ Z assoziiert.
- Die Funktion zugeordnete_zugriffsrechte (ro:Ro) $\rightarrow$ 2^Z beschreibt die einer Rolle zugeordnete Menge an Zugriffsrechten: zugeordnete_zugriffsrechte (ro) = { z $\in$ Z | (z, ro) $\in$ ZZ}.
- Op (z:Z) $\rightarrow$ {op$\subseteq$ Op} ist die Zugriffsrecht-Operatoren Zuordnung, die eine Menge an Operatoren ausgibt, welche einem Zugriffsrecht zugeordnet sind.
- O (z:Z) $\rightarrow$ {o$\subseteq$ O} ist die Zugriffsrecht-Objekt Zuordnung, die eine Menge an Objekten ausgibt, die einem Zugriffsrecht zugeordnet sind.
- subjekt_sitzung S $\rightarrow$ 2^{Si}, die Abbildung von einem Subjekt s $\in$ S auf eine Menge von Sitzungen si $\in$ Si.
- sitzung_subjekt (si:Si) $\rightarrow$ 2^S, die Abbildung von einer Sitzung si $\in$ Si auf das korrespondierende Subjekt s $\in$ S.
- sitzung_rollen Si $\rightarrow$ 2^{Ro}, die Abbildung von einer Sitzung si $\in$ Si auf eine Menge von Rollen. sitzung_rollen (si) $\subseteq$ {ro $\in$ Ro | (sitzung_subjekt (si), ro) $\in$ ZZ} bezeichnet die aktivierten Rollen einer Sitzung.

[57] Die Rollentypen werden durch die Aufteilung in Rollen, Administrationsrollen etc. implizit in der Grafik dargestellt.

– verfügbare_sitzung_zugriffsrechte(si:Si) $\rightarrow 2^{Z}$, in einer Sitzung si $\in$ Si verfügbaren Zugriffsrechte eines Subjektes.
verfügbare_sitzung_zugriffsrechte(si) = {ro $\in$ Ro | $\exists$ sitzung_rollen (si) $\wedge$ $\exists$ zugeordnete_zugriffsrechte (ro) }

Rollenhierarchie

– RH $\subseteq$ Ro $\times$ Ro ist eine partielle Ordnung auf Rollen, die Vererbungsrelation genannt wird. Geschrieben „$\geq$“, wobei $ro_1 \geq ro_2$ nur gilt, wenn alle Zugriffsrechte von ro_2 auch die Zugriffsrechte von ro_1. Zum Beispiel: $ro_1 \geq ro_2$ $\Rightarrow$ autorisierte_zugriffsrechte(ro_2) $\subseteq$ autorisierte_zugriffsrechte(ro_1).

– autorisierte_subjekte (ro:Ro) $\rightarrow 2^{S}$, die Zuordnung von Rollen auf eine Menge von Subjekten in Berücksichtigung der Rollenhierarchie.
autorisierte_subjekte (ro) = {s $\in$ S | ro' $\geq$ ro, (s, ro' $\in$ SZ}

– autorisierte_zugriffsrechte(ro:Ro) $\rightarrow 2^{Z}$, die Zuordnung von Rollen auf eine Menge von Zugriffsrechten unter Berücksichtigung der Rollenhierarchie.
autorisierte_zugriffsrechte (ro) = {z $\in$ Z | ro' $\geq$ ro, (z, ro' $\in$ ZZ}

Aufgabentrennung

– SAT $\subseteq (2^{RO})$ ist eine Sammlung von Tupeln (rom) in statischer Aufgabentrennung (SAT), eine Menge von Rollen (rom) und eine Teilmenge (t) von Rollen aus der Menge rom, mit der Eigenschaft, dass ein Subjekt nur zu einer Rolle aus der Menge rom in jedem sat $\in$ SAT zugeordnet werden kann.

$$\forall \text{ (rom) } \in \text{SAT}, \forall t \subseteq \text{rom} : |t| = 1 \Rightarrow \bigcap_{ro \in t} \textit{subjektzuordnung} \text{ (ro)} = \varnothing$$

– Bei einer existierenden Rollenhierarchie wird die statische Aufgabentrennung auf *autorisierte_subjekte* anstatt auf *subjektzuordnung* wie folgt neu definiert.

$$\forall \text{ rom} \in \text{SAT } \forall t \subseteq \text{rom} : |t| = 1 \Rightarrow \bigcap_{ro \in t} \text{autorisierte\_subjekte (ro)} = \varnothing$$

– DAT $\subseteq (2^{RO})$ ist eine Sammlung von Paaren (rom) mit dynamischer Aufgabentrennung, in dem jedes rom eine Rollenmenge darstellt, mit der Eigenschaft, dass ein Subjekt nur eine Rolle aus der Menge rom in jedem dat $\in$ DAT aktivieren kann.

$\forall$ rom $\in 2^{RO}$, $\forall$ (rom) $\in$ DAT und

$\forall$ si $\in 2^{Si}$, $\forall$ rom $\in 2^{RO}$, $\forall$ rollen_teilmenge $\in 2^{RO}$, (rom) $\in$ DAT,

rollen_teilmenge $\subseteq$ rom, *rollen_teilmenge* $\subseteq$ *sitzung_rollen* (si) $\Rightarrow$ | *rollen_teilmenge* | = 1.

Administration

- AZ = $2^{O \times Op}$, eine Menge Administrationszugriffsrechten, es gilt $\forall$ az $\in$ AZ az $\notin$ Z.
- ARo $\cap$ Ro = $\varnothing$
- *ZP = 2^Z, eine Menge von Zugriffsrechten, die zu einem Zugriffsrechtepool zusammengefasst sind. ZP wird bei der Administration ausgewertet.*
- *SP = 2^P, eine Menge von Subjekten, die zu einem Subjektpool zusammengefasst sind. SP wird bei der Administration ausgewertet.*

Eine Vorbedingung ist ein Boolescher Ausdruck unter der Verwendung des logischen UND-Operators „$\wedge$“ oder dem logischen ODER-Operators „$\vee$“. Die Operatoren werden auf den Termen der Form s und $\overline{s}$, wobei s $\in$ SP und z und $\overline{z}$, wobei z $\in$ ZP angewendet. Die Vorbedingung wird ausgewertet im Bereich des Subjektpools (SP) in Bezug auf ein bestimmtes Subjekt s $\in$ S oder beim Zugriffsrechtspool (ZP) auf ein bestimmtes Zugriffsrecht z $\in$ Z.

- Die Vorbedingung wird für ein Subjekt s bezüglich des Terms s als wahr interpretiert, wenn gilt: $\exists$ s $\in$ SP.
- Die Vorbedingung wird für ein Subjekt s bezüglich des Terms $\overline{s}$ als wahr interpretiert, wenn gilt: $\exists$ s $\notin$ SP.
- Die Vorbedingung wird für ein Zugriffsrecht z bezüglich des Terms z als wahr interpretiert, wenn gilt: $\exists$ z $\in$ ZP.
- Die Vorbedingung wird für ein Zugriffsrecht z bezüglich des Terms $\overline{z}$ als wahr interpretiert, wenn gilt: $\exists$ z $\notin$ ZP.

Ein Rollenbereich beschreibt eine Menge von Rollen mit der Bereichsnotation einer oberen ro_o und unteren ro_u Grenze. Es gilt:

- $[ro_u, ro_o] = \{ ro \in Ro \mid ro_u \leq ro \wedge ro \leq ro_o\}$ $[ro_u, ro_o) = \{ ro \in Ro \mid ro_u \leq ro \wedge ro < ro_o\}$
- $(ro_u, ro_o] = \{ ro \in Ro \mid ro_u < ro \wedge ro \leq ro_o\}$ $(ro_u, ro_o) = \{ ro \in Ro \mid ro_u < ro \wedge ro < ro_o\}$

Alle möglichen Rollenbereiche bei einer gegebenen Rollenhierarchie werden durch die Menge der Rollenbereiche ROB beschrieben: ROB $\rightarrow$ 2^{Ro}.

Delegation

- $Z^D = 2^{O \times Op}$, *eine Menge von Zugriffsrechten die delegierbar sind, es sind* $z^D \subseteq Z \wedge (z^D, ro) \in ZZ \wedge (s,ro) \in SZ$.
- $DRo \cap ARo \cap Ro = \varnothing$

Die Operatoren werden auf den Termen der Form ro und $\overline{ro}$ angewendet, wobei ro $\in$ Ro. Die Vorbedingung wird ausgewertet im Bereich der Zugriffsrechtszuordnung ZZ auf ein bestimmtes Zugriffsrecht z $\in$ Z und der Subjektzuordnung SZ in Bezug auf ein bestimmtes Subjekt s $\in$ S für die Erzeugung einer Delegationsrolle und der dabei notwendigen Zuordnung von Zugriffsrechten zu der Delegationsrolle.

- Die Vorbedingung wird für ein Zugriffsrecht z bezüglich des Terms ro als wahr interpretiert, wenn gilt: $\exists$ (ro' $\geq$ ro) (z, ro') $\in$ ZZ und wenn für Subjekt s gilt: $\exists$ (ro' $\geq$ ro) (s, ro') $\in$ SZ.
- Die Vorbedingung wird für ein Zugriffsrecht z bezüglich des Terms $\overline{ro}$ als wahr interpretiert, wenn gilt: $\exists$ (ro' $\geq$ ro) (z, ro') $\notin$ ZZ.
- DRo $\notin$ RH; eine Delegationsrolle wird in keine Rollenhierarchie eingeordnet.

Personalisierung durch Schlüsselwerte

- $Ro^{Sw} \subseteq Ro$, *personalisierbare Rollen ro sind eine Teilmenge der Rollen, deren Attribut* ***Datenobjekt*** *sich auf ein Objekt aus dem Ziel-AwS bezieht. Eine Rolle* $ro^{Sw} \in Ro^{Sw}$ *bezieht sich auf genau ein Datenobjekt.*
- $SZSw \subseteq SZ \times Sw$, *eine 1:n Zuordnungsrelation von Schlüsselwerten Sw zu einer Subjektzuordnung SZ. Über ein Tupel (sw, sz) werden mit einer Subjektzuordnung* $sz \in SZ$ *die möglichen Ausprägungen der* $sw \in Sw$ *assoziiert.*

Parametrisierung

- *PT = ein Parametertyp* $pt \in PT$ *bezieht sich auf ein beliebiges Objekt aus dem Ziel-AwS.*
- $Ro^P \subseteq Ro$, *parametrisierbare Rollen ro sind eine Teilmenge der Rollen. Eine Rolle* $ro^P \in Ro^P$ *bezieht sich auf beliebig viele Objekte aus dem Ziel-AwS.*
- $SZ^PSw \subseteq SZ \times PT \times Sw$, *eine 1:n Zuordnungsrelation zwischen einer Subjektzuordnung für einen Parametertyp PT mit den jeweiligen Schlüsselwerten aus dem Ziel-AwS. Über ein Tupel (sz,pt,sw) werden für eine Subjektzuordnung* $sz \in SZ$ *mit einem Parametertyp* $pt \in PT$ *Schlüsselwerte aus dem AwS assoziiert.*

6.10 Zusammenfassung

Das Zugriffskontrollmodell eRBAC erweitert das Referenzmodell RBAC. Besonders hervorzuheben sind neben der modifizierten Übernahme der Konzepte aus Kapitel 4 die Einführung von Objekttypen, Rollentypen, Personalisierung und Parametrisierung. Durch die Einführung von Objekttypen zur Strukturierung der Objekte kann zwischen dem Aufruf einer Anwendungssystems und der Autorisierung innerhalb eines Anwendungssystems unterschieden. Dies bildet die Grundlage für die Implementierung einer flexiblen Aufrufstruktur. Die möglichen Rollentypen sind: Anwendungsrollen, Administrationsrollen, virtuelle Rollen und Delegationsrollen. Die Einteilung der Rollen in Rollentypen ermöglicht getrennte Sicherheitsrichtlinien und erleichtert die Administration. Dadurch können u.a. Fehler bei der Subjektzuordnung minimiert werden. Virtuelle Rollen sorgen dahingegen für eine überschaubare Rollenhierarchie. Durch die Verbindung des Zugriffskontrollmodells mit den Schlüsselattributen aus dem Zielanwendungssystem wird die Personalisierung und Domänenbeschränkung umgesetzt. So kann durch die Personalisierung der Subjektzuordnung pro Aufgabenträger z. B. mit der Rolle LM[58] auf mehrere Lehrstühle zur Bearbeitung zugegriffen werden. Die Domänenbeschränkung sorgt dafür, dass die Anzahl der Rollen überschaubar bleibt. Damit kann realisiert werden, dass Aufgabenträgern derselben Rolle unterschiedliche Domänen als Ausgabedaten in Abhängigkeit von Parametern ausgeliefert werden.

Für die Umsetzung der Zugriffskontrolle müssen die drei Abstraktionsebenen (siehe Kapitel 2.4.1): Zugriffskontrollstrategie, Zugriffskontrollmodell und Zugriffskontrollmechanismus betrachtet werden. Die Zugriffskontrollstrategie wird durch die Sicherheitsstrategie des Unternehmens festgelegt. Für die Umsetzung der Zugriffskontrollstrategie muss ein entsprechendes Zugriffskontrollmodell gewählt werden. Das Zugriffskontrollmodell eRBAC erfüllt eine rollenbasierte Zugriffskontrollstrategie (siehe Kapitel 3.2.7.7). eRBAC ist das zugrundeliegende Zugriffskontrollmodell für den exemplarisch realisierten Zugriffskontrollmechanismus FN2RBAC, der im Folgenden beschrieben wird.

[58] LM steht für das Lehrstuhlmodul in FlexNow und wurde als Rollenbezeichnung übernommen, da damit alle Aufgaben eines Lehrstuhls durchgeführt werden können.

7 Exemplarische Realisierung von eRBAC für FlexNow

Nach der konzeptionellen und formalen Beschreibung des Zugriffskontrollmodells eRBAC wird auf dessen Grundlage mit dem Zugriffskontrollsystem FN2RBAC ein exemplarisch realisierter Zugriffskontrollmechanismus für die Zugriffskontrolle von FlexNow beschrieben. FN2RBAC beinhaltet die Rechteverwaltung (FN2RBAC-V), die Rechteprüfung (FN2RBAC-RP) und Protokollierung (FN2RBAC-P). Neben der Architektur wird die Konzeption und Funktionsweise in diesem Kapitel vorgestellt. Das Konzept der Delegation wurde in das Zugriffskontrollmodell eRBAC aufgenommen, jedoch noch nicht in FN2RBAC umgesetzt.

Da eine Authentifizierung einer Zugriffskontrolle zwingend vorgelagert ist (siehe Kapitel 2.3), wird innerhalb dieses Kapitels neben FN2RBAC auch das Authentifizierungsportal FN2AUTH für webbasierte Anwendungssysteme beschrieben. Die zunächst vorgestellte Architektur von FlexNow2 (FN2), die Neu- und Weiterentwicklung des Prüfungsverwaltungssystems FlexNow, ist auch die Grundlage für die Architektur des Authentifizierungsportals (FN2AUTH) und des Zugriffskontrollsystems (FN2RBAC).

7.1 Architektur von FN2

Die verwendete Programmiersprache für die Neuentwicklung von FlexNow: FN2, FN2AUTH und FN2RBAC ist Java. Als Architektur wird die Java™ Platform Enterprise Edition (JEE) (Oracle Corporation) eingesetzt.

Die umgesetzte Web-Architektur von FN2 ist eine mehrschichtige Client-Server-Architektur mit einer Client-Schicht und mehreren Schichten von Servern. Im Weiteren wird von der Clientschicht, die durch einen Web-Browser realisiert wird, abstrahiert. Auf der Serverseite lassen sich maximal drei Serverschichten unterscheiden (Ferstl und Sinz 2013, S. 493–495):

- Web-Server: Verteilung der Dokumente, die vom Client angefordert werden
- Anwendungsserver: Bereitstellung und Steuerung der Objekte zur Realisierung von Anwendungsfunktionen
- Daten-Server: Verwaltung der persistenten Daten

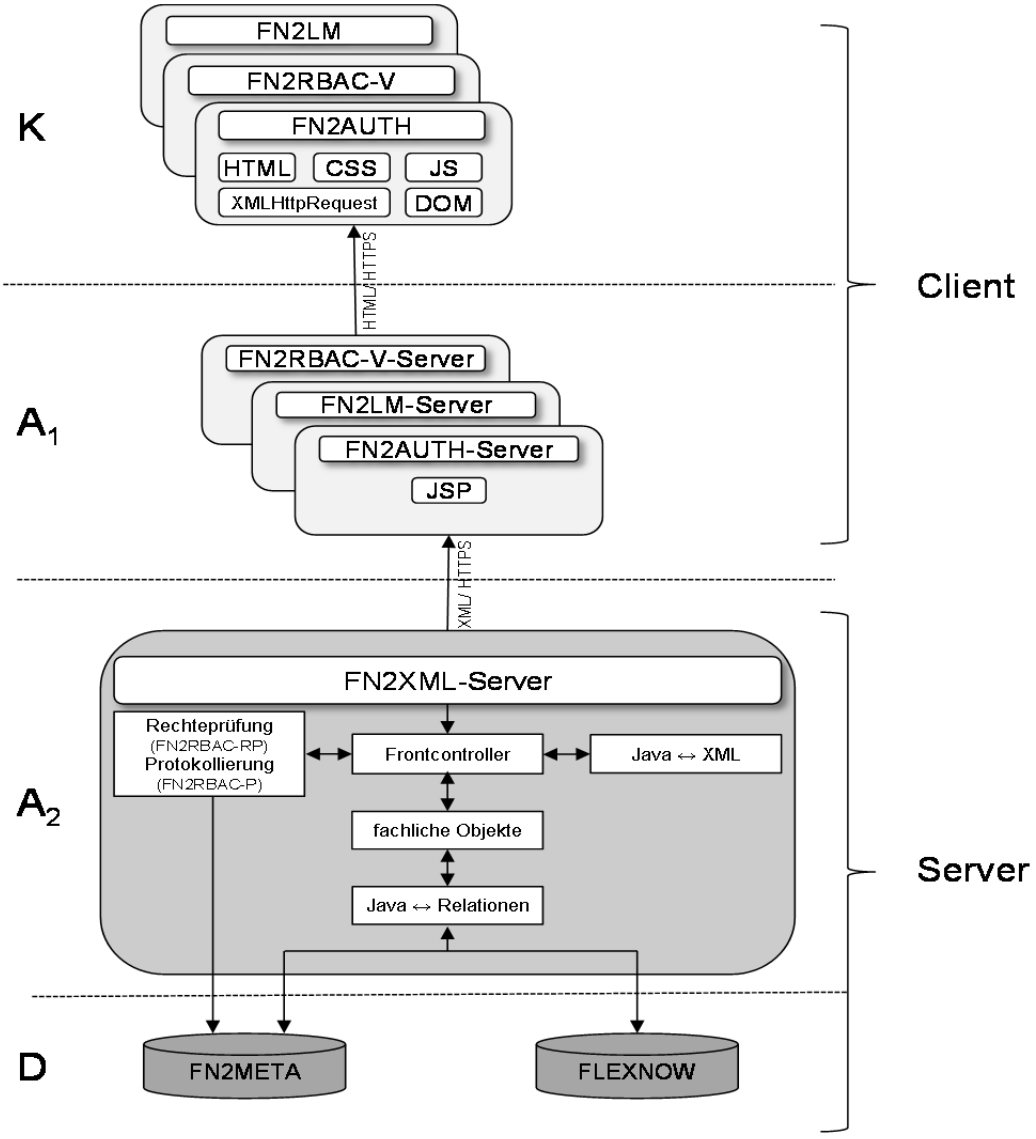

Abb. 7-1 zeigt die Realisierung der Client-Server-Architektur in FN2:

Die drei Serverschichten sind in FN2, wie in **Abb. 7-1** dargestellt, folgendermaßen realisiert:

- Die zentrale Komponente der Architektur ist der FN2XML-Server. Dieser kapselt die fachliche Logik des Anwendungssystems FlexNow, des Authentifizierungsportals FN2AUTH sowie des Zugriffskontrollsystems FN2RBAC und kann über HTTP bzw. HTTPS Protokolle angesprochen werden. Der FN2XML-Server ist Teil des Anwendungsservers (A_2).
- Die FN2Klienten[59] beinhalten einen Kommunikationsteil und einen Anwendungsteil. Im Anwendungsteil werden die XML-Dokumente für Anfragen

[59] FN2Klient ist der Client und ein Sammelbegriff für die einzelnen Anwendungen von FlexNow2: Bspw. FN2LM für Lehrstühle oder FN2STUD für Studierende.

aufbereitet und an den FN2XML-Server geschickt. Anschließend wird die zurückgelieferte XML-Antwort entgegen genommen. Die Antwort wird bei Bedarf im Anwendungsteil bearbeitet und anschließend im Kommunikationsteil die Darstellung entsprechend aufbereitet und an den Web-Browser geschickt. Der Kommunikationsteil der FN2Klienten wird durch den Web-Server (K) umgesetzt. Der Anwendungsteil ist Teil des Anwendungsserver (A_1).

- Die beiden Datenbankinstanzen FLEXNOW und FN2META persistieren die Daten und werden durch den Daten-Server (D) verwaltet. In der Datenbankinstanz FLEXNOW werden alle Informationen zum Prüfungsverwaltungssystem FlexNow und in FN2META alle Daten des Authentifizierungssystems FN2AUTH und Zugriffskontrollsystems FN2RBAC gespeichert.

Nach dem ADK-Strukturmodell[60] gliedert sich die Architektur von FN2 wie folgt:

- Anwendungsfunktionen (A) werden im Serverteil vom jeweiligen FN2Klient (A_1) und vom FN2XML-Server (A_2) realisiert.
- Die Datenverwaltung (D) wird von den Datenbankinstanzen FLEXNOW und FN2META übernommen.
- Der Kommunikationsbereich (K) befindet sich zum einem im Web-Browser zum anderen in der Clientschicht der FN2Klienten.

FN2XML-Server: Aufbau und Funktionsweise

Der FN2XML-Server fungiert als Datenquelle, liefert Informationen in Form von XML-Dokumenten an die Clients aus und ist als Webservice[61] ansprechbar. Es wurde das Paradigma des Webservice gewählt, da der Austausch von Informationen mittels XML-Dokumenten sowie die Verwendung von Standardprotokollen wie HTTP bzw. HTTPS eine „zuverlässige und kostengünstige Integration von verteilten Softwaresystemen“ erlauben (Stiemerling 2002, S. 435). Zudem können Webservices als ein Paradigma für verteilte Anwendungen zu einer verbesserten Wiederverwertung und Interoperabilität führen (Feng et al. 2004, S. 357). Anfragen an den FN2XML-Server erfolgen nicht mittels des Simple Objekt Access Protocols (SOAP)

[60] Für die Beschreibung des ADK-Modells wird auf Ferstl und Sinz (2013, S. 321–323) verwiesen.
[61] Ein Webservice wird von einem Webserver als Dienst zur Verfügung gestellt, dessen Ausgabe über eine standardisierte Schnittstelle von Computern gelesen und bedient werden kann Stiemerling (2002, S. 435); W3C (2009).

(Box et al. 2000), sondern orientieren sich eher an dem Representational State Transfer (REST) Protokoll (Fielding 2002)[62].

Das HTTP bzw. HTTPS Protokoll kennt zwei Arten zur Übermittlung von Anfragen an einen Webserver: **post**- bzw. **get**-Aufrufe[63]. Die Anfragen an den FN2XML-Server erfolgen normalerweise mittels **post**-Aufruf, in dessen Body ein XML-Dokument mitgegeben wird. Für die Anzeige von Listenelementen steht die einfachere Variante der **get**-Aufrufe zur Verfügung. Die Ausgabeformate des FN2XML-Servers sind XML-Dokumente oder binäre Dateien (z. B: PDF, CSV).

Der FN2XML-Server hat folgenden Aufbau (siehe **Abb. 7-1**):

- FrontController: Entgegennahme aller Anfragen von FN2Klienten.
- Rechteprüfung und Protokollierung: Jeder Zugriff wird zuerst überprüft, ob dieser erlaubt ist und anschließend protokolliert.
- Java ↔XML: Diese Komponente hat zwei Funktionen: Zum einen werden aus einem XML-Dokument Java-Objekte erzeugt, zum anderen werden für die FN2Klienten Java-Objekte zu XML-Dokumenten aufbereitet.
- Fachliche Objekte: In den fachlichen Objekten steckt die Geschäftslogik von FN2 und umfasst sowohl konzeptionelle Objekte als auch Vorgangsobjekte.
- Java ↔ Relationen: Umwandlung der Java-Objekte auf die entsprechenden Relationen in der relationalen Datenbank mit Hilfe eines Objekt-Relationen-Mapping[64].

Der Prozess der Bearbeitung einer Anfrage an den FN2XML-Server läuft wie folgt ab: Der FrontController nimmt alle Anfragen der FN2Klienten entgegen. Vor der Transformation ankommender XML-Dokumente in fachliche Java-Objekte wird zunächst überprüft, ob eine gültige Sitzung existiert. Danach wird die Erlaubnis des Zugriffs auf diese im XML angeforderten Objekte mit den dazugehörigen Operatoren mit FN2RBAC-RP kontrolliert und mit FN2RBAC-P protokolliert. Nach erfolgreicher Autorisierung wird die Anfrage fachlich bearbeitet und durch entsprechende Abfragen über Objekt-Relationen-Mapping an die Datenbank werden die benötigten Daten geholt. Dabei werden, falls notwendig, Informationen in der entsprechenden

[62] Für ein vertiefendes Studium der Funktionsweise der beiden Protokolle wird auf die angegebene Literatur verwiesen.
[63] Die ausführliche Spezifikation ist zu finden unter W3C (1999).
[64] Das Objekt-Relationen-Mapping wird von Hibernate zur Verfügung gestellt o.V. (2014b).

Datenbank gespeichert. Nachdem alle notwendigen Informationen bereit stehen, werden aus Java-Objekten XML-Dokumente erzeugt, bei Bedarf in PDF oder CSV umgewandelt und an den aufrufenden Client ausgeliefert.

7.2 Authentifizierungsportal FN2AUTH

Das Authentifizierungsportal FN2AUTH beinhaltet neben der kompletten Abwicklung der Authentifizierung auch die Funktionen der Zugriffskontrolle, die für allen Clients benötigt werden. Diese Funktionen werden unter dem Begriff anwendungsübergreifenden Teil der Zugriffskontrolle zusammengefasst. Für die Umsetzung der Authentifizierung wird das Konzept der Authentifizierungstypen verwendet. Ein Authentifizierungstyp beschreibt die Kombination aus einem Authentifizierungsmerkmal und einem Verfahren, mit dem dieses Merkmal überprüft wird. Für den anwendungsübergreifenden Teil der Zugriffskontrolle werden die Konzepte Objekttyp (siehe Kapitel 6.1.1) und Personalisierung (siehe Kapitel 6.7) aus eRBAC verwendet.

Um eine Authentifizierung für verschiedene Authentifizierungstypen und beliebige Clients effizient implementieren zu können, wurde eine eigenständige Webanwendung als Portal (FN2AUTH) gewählt. Das Authentifizierungsportal FN2AUTH kann optional auf unterschiedlichen Applikationsservern für Intranet, Extranet oder Internet mit jeweils unterschiedlichen Authentifizierungstypen installiert werden. In Abhängigkeit von den Rollen kann festgelegt werden, welcher Nutzerkreis sich über die jeweilige Installation von FN2AUTH authentifizieren darf.

> *Beispielsweise authentifizieren sich Studierende über LDAP und Mitarbeiter der Prüfungsämter über die in einer Datenbank gespeicherten Zugangsdaten der Subjekte. Es kann dadurch z. B. realisiert werden, dass Studierende und Prüfungsamtsmitarbeiter unterschiedliche Webadressen zum Anmelden erhalten.*

Die Aufrufstruktur des Authentifizierungsportals wird in Abb. 7-2 dargestellt. Der erste Aufruf erfolgt immer über FN2AUTH. Sollte ein FN2Klient direkt aufgerufen werden und es liegt noch keine Authentifizierung vor, wird an FN2AUTH weitergeleitet. Nach einer erfolgreichen Authentifizierung wird die Information der Sitzung an den jeweiligen Client weitergegeben. Danach erfolgt die Kommunikation immer

direkt zwischen dem jeweiligen FN2Klienten und dem FN2XML-Server (siehe Kapitel 7.1).

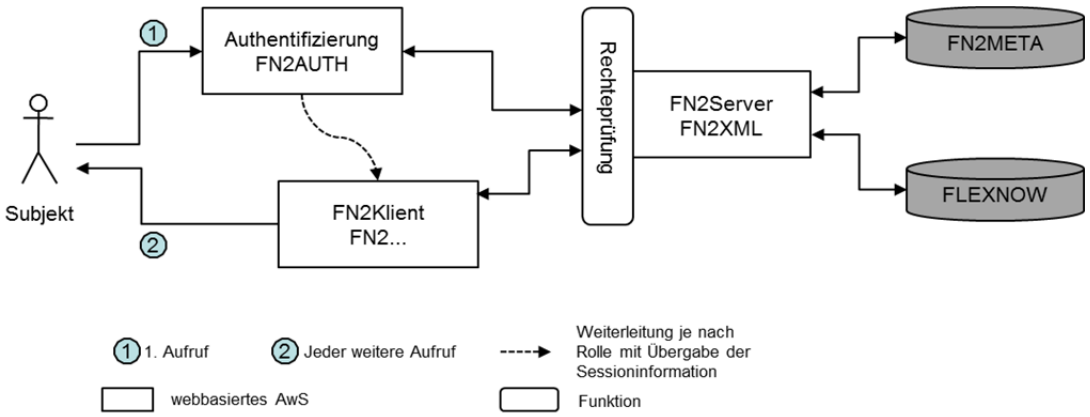

Abb. 7-2 Aufrufstruktur des Authentifizierungsportals FN2AUTH

FN2AUTH deckt folgende Funktionen ab:

- Zugangsdaten anfordern, wenn noch keine Authentifizierung vorgenommen wurde (1. Aufruf).
- Eingegebene Zugangsdaten werden je nach Authentifizierungstyp überprüft.
- Die Authentifizierung protokollieren.
- Anwendungsübergreifend werden in Abhängigkeit der zugeordneten Rollen die autorisierten Anwendungen zur Auswahl gestellt sowie anschließend die Personalisierung vorgenommen.

7.2.1 Authentifizierung

Jedes Unternehmen und jede Verwaltung verfügt über eine eigene Infrastruktur für Informationstechnik und Informationssicherheit. Die Speichermöglichkeiten der Zugangsdaten reichen von einer Datenbank über LDAP (Kapitel A.1.1) bis hin zu Chipkarten (Kapitel 2.3.3). Durch das Konzept der Authentifizierungstypen werden diese verschiedenen Authentifizierungsmöglichkeiten unterstützt.

Es kann für jede Installation ein Authentifizierungstyp voreingestellt werden, z. B. LDAP. FN2AUTH überprüft zuerst diesen voreingestellten Authentifizierungstyp. Sind einem Subjekt noch weitere Authentifizierungstypen zugeordnet, werden diese nach dem Scheitern des Versuches mit dem voreingestellten Typ aufgerufen und überprüft, ob eine Authentifizierung mit einem anderen Typ erfolgreich ist. Dürfen sich Subjekte ausschließlich mit dem voreingestellten Authentifizierungstyp authentifizieren, so wird diesen nur dieser Authentifizierungstyp zugeordnet.

Damit besteht die Flexibilität, dass beispielsweise alle Prüfungsamtsmitarbeiter sich über LDAP authentifizieren, aber ein Supportmitarbeiter, der kein Login im LDAP-Verzeichnis hat, sich über eine Datenbank authentifizieren kann.

7.2.2 Anwendungsübergreifende Autorisierung und Personalisierung

Nach erfolgter Authentifizierung werden durch FN2RBAC mit Hilfe des Konzeptes der Objekttypen die autorisierten Anwendungen ermittelt. Sind einem Subjekt mehrere Rollen mit je einem Objekt des Typs **Anwendung** oder eine Rolle mit mehreren Objekten des Typs **Anwendung** zugeordnet, wird dem Subjekt nach der Zugriffskontrolle eine Auswahlliste möglicher Anwendungen zur Verfügung gestellt. Ist einem Subjekt nur ein Objekt des Typs **Anwendung** zugeordnet, kann FN2AUTH das Subjekt automatisch an die jeweilige Anwendung weiterleiten, ohne dass das Subjekt eine Auswahl treffen muss.

Nach der Auswahl der gewünschten Anwendung wird falls erforderlich die Personalisierung durch das Zugriffskontrollsystem FN2RBAC für FN2AUTH vorgenommen. Es wird ermittelt, ob dem authentifizierten Subjekt für die gewählte Anwendung mehr als ein Eintrag an Schlüsselattributen zugeordnet ist. Ist dies der Fall, wird wieder eine Auswahlliste angeboten, um z. B. die entsprechende Organisationseinheit auswählen zu können. Ist nur eine Organisationseinheit zugeordnet, erfolgt ebenfalls eine automatische Weiterleitung.

7.3 Zugriffskontrolle mit eRBAC: FN2RBAC

FN2RBAC ist die Implementierung des Zugriffskontrollmechanismus für das Zugriffskontrollmodell eRBAC (siehe Kapitel 6). FN2RBAC besteht aus folgenden drei Bereichen:

- FN2RBAC-V: Ein eigenständiger Client für die Administration.
- FN2RBAC-RP: Die Rechteprüfung ist als Referenzmonitor, wie im Standard ISO 10181-3 vorgeschrieben implementiert.
- FN2RBAC-P: Jeder Zugriff, ob dieser erlaubt ist oder nicht, wird protokolliert. Die Protokollierung ist direkt im FN2XML-Server integriert.

In FN2RBAC wurden neben der Zugriffskontrolle konkreter Subjekte auch Konzepte für anonyme Zugriffe mit Gastrollen und dazugehörigen Gastnutzern konzipiert. Für

den Aufruf von Gastrollen wurden zwei unterschiedliche Konzepte implementiert, je nachdem, ob die Zugriffe protokolliert werden müssen oder nicht: Gastaccount und immerwährende Session.

Müssen die Zugriffe protokolliert werden, dann wird durch die Übergabe eines Parameters ein Gastaccount aktiviert. Dazu sind in einer Konfigurationsdatei des Applikationsservers die Zugangsdaten zu hinterlegen. Für das Subjekt erfolgt die Authentifizierung transparent.

> *Beispielsweise kann ein Studierender in FlexNow einen überprüfbaren Leistungsnachweis als PDF erzeugen. Dieser ist mit ID und Kennwort versehen und liegt z. B. einer Bewerbung bei. Nun kann ein Personalchef auf einer Internetseite der ausstellenden Hochschule durch die Eingabe von ID und Kennwort des Leistungsnachweises genau diesen Leistungsnachweis aufrufen. Es ist keine individuelle Authentifizierung erforderlich, sondern es wird ein Gastaccount mit entsprechender Gastrolle mit den notwendigen Zugriffsrechten aktiviert.*

Muss der Zugriff nicht protokolliert werden, kann das Konzept einer immerwährenden Session gewählt werden. Dazu wird die sessionID der immerwährenden Session beim Aufruf übergeben. Eine Authentifizierung ist dadurch nicht erforderlich. Dieser immerwährenden Session ist der entsprechende Gastnutzer zugeordnet. Dieser Gastnutzer besitzt genau eine Gastrolle mit den entsprechenden wenigen Zugriffsrechten. Anwendung findet dieses Konzept bei der Übergabe der Zugangsdaten, während des Anmeldevorgangs, um eine neue Sitzung zu erzeugen. Der Nutzer hat zu diesem Zeitpunkt noch keine authentifizierte Sitzung aufgebaut. Anonyme Nutzer können außer über die aufrufende IP nicht zurückverfolgt werden und eine Protokollierung der Zugriffe durch FN2AUTH ist dabei nicht weiter hilfreich. Die Client-IP wird z. B. bereits durch die Protokoll-Mechanismen der Web-Server bzw. Anwendungsserver bereitgestellt.

7.3.1 Administration von eRBAC: FN2RBAC-V

Die Administration (FN2RBAC-V) besteht aus folgenden vier Bereichen:

- Grundlagen,
- Rollenverwaltung,
- Subjektverwaltung

- und Domänenverwaltung.

Diese Bereiche werden in **Abb. 7-3** dargestellt und können an unterschiedliche Administratoren vergeben werden. Damit kann eine Dezentralisierung der Administration erreicht werden. Für eine Verwaltung der Administratoren kann ebenfalls FN2RBAC verwendet werden, indem für jeden Bereich eine eigene Rolle angelegt und mit den entsprechenden Zugriffsrechten versehen wird.

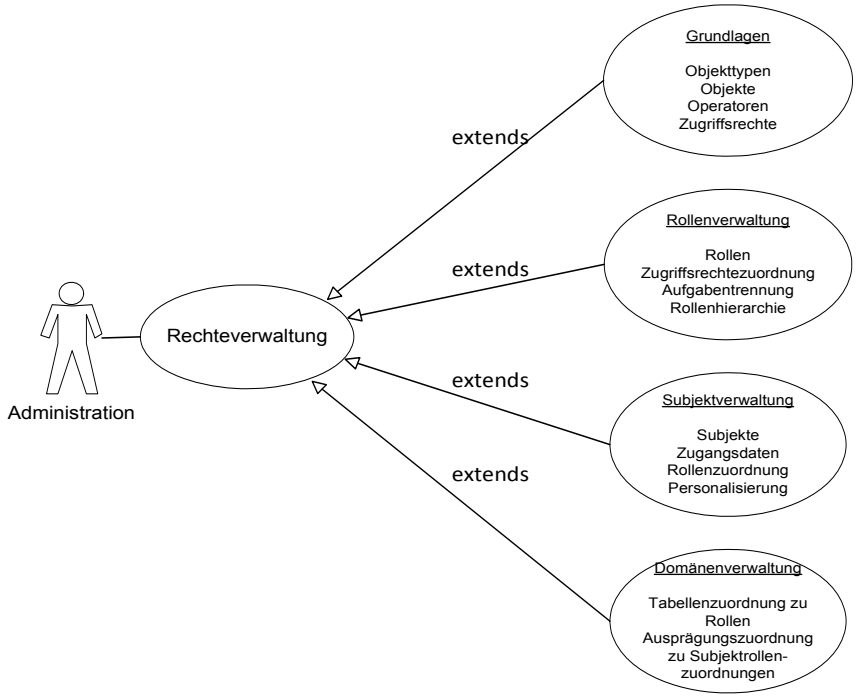

Abb. 7-3 Anwendungsfall Rechteverwaltung

Die Objekt- und Rollentypen werden bei der Installation festgelegt und müssen später nicht mehr bearbeitet werden. Der Funktionsbereich der **Grundlagen** umfasst das Anlegen von neuen Objekten, Operatoren und die Zuordnung von Objekten und Operatoren zu erlaubten Zugriffsrechten.

In der **Rollenverwaltung** werden Rollen angelegt und Zugriffsrechte den jeweiligen Rollen zugeordnet. Nach dem Anlegen einer Rolle muss festgelegt werden, ob eine Aufgabentrennung zu bereits existierenden Rollen besteht und ob diese statisch oder dynamisch zu definieren ist. Danach muss die neue Rolle in die Rollenhierarchie eingeordnet werden.

Die **Subjektverwaltung** umfasst das Bearbeiten von Subjekten einschließlich der Zugangsdaten und das Zuordnen von Rollen zu diesen Subjekten. Nach der Sub-

jektzuordnung muss bei Bedarf die Personalisierung vorgenommen werden, um das Subjekt für die notwendigen Informationen freizuschalten.

Die **Domänenverwaltung** umfasst die Zuordnung von Parametern, die sich auf den Tabellennamen aus dem Zielanwendungssystem beziehen, zu Rollen, für die eine Domänenbeschränkung notwendig ist. Nach der Zuordnung der Parameter müssen für jedes Subjekt, das einer parametrisierbaren Rolle zugeordnet wird, die erlaubten Ausprägungen der Attributwerte zugeordnet werden.

7.3.2 Rechteprüfung in eRBAC: FN2RBAC-RP

Nach erfolgter Authentifizierung muss jeder Zugriff autorisiert werden. Die Implementierung der Rechteprüfung und Protokollierung wurde nach dem Konzept ISO Access Control Framework als dritter Teil des Standards ISO 10181-3 (Biltzinger und Bunz 2004, S. 31) als Referenzmonitors (Anderson 1972, S. 24) umgesetzt. Durch die Implementierung an zentraler Stelle wird vermieden, das gesamte Anwendungssystem mit Durchsetzungsfunktionen (siehe Kapitel 2.4.4) zu durchziehen. „Eine weitgehende Separation der Überprüfung der Autorisierung vom Anwendungscode ermöglicht es, Anwendungscode zu schreiben, der nicht mit Autorisierungscode durchsetzt ist“ (Riechmann 1999, S. 3). Dies sollte vor allem in verteilten objektorientierten Systemen berücksichtigt werden (Riechmann 1999, S. 3). Die Rechteprüfung und Protokollierung wurde auf zwei Arten implementiert:

- Integration als Klasse im FN2XML-Server als Referenzmonitor
- Von außen ansprechbarer Webservice des FN2XML-Servers.

Der FN2XML-Server (siehe **Abb. 7-1**) wird als zentrale Komponente von allen FN2Klienten aufgerufen. Durch diesen zentralen Server kann die Rechteprüfung als Integration im FN2XML-Server vorgenommen werden. Wodurch der Implementierungsvorschlag für die Rechteprüfung als zentrale Durchsetzungsfunktion des ISO Access Frameworks und Referenzmonitors umgesetzt wurde.

Die Klasse **Rechteprüfung** wird vom FrontController in FN2XML genutzt, um die Rechteprüfung vorzunehmen. Die Übergabeparameter sind die SessionID und das Objekt mit seinem zugehörigen Operator. Der FrontController überprüft, ob eine gültige Session vorhanden ist und ruft anschließend die Klasse zur Rechteprüfung auf. Anhand der sessionID werden das Subjekt und die aktivierte Rolle ermittelt. Danach wird anhand des aufzurufenden Objektes und Operators geprüft, ob der Auf-

ruf erlaubt ist. Es kann also kein Zugriff zur Datenbank über FN2XML erfolgen, ohne dass durch den Referenzmonitor überprüft wurde, ob dieser Zugriff erlaubt ist. Nach erfolgter Zugriffskontrolle wird protokolliert, welches Subjekt welches Objekt mit welchem Operator aufrufen will.

Die Funktion der Rechteprüfung steht auch als Webservice zur Verfügung, damit andere Webanwendung FN2RBAC für Zugriffskontrollinformationen nutzen können. Hierfür wird ein XML-Dokument, in dem das Subjekt, die aktive Rolle, das Objekt und der Operator beschrieben werden, an den FN2XML-Server geschickt. Die Klasse **Rechteprüfung** ermittelt, ob ein Zugriffsrecht vorhanden ist und liefert das Ergebnis zurück. Es wird anschließend vom FN2XML-Server an die aufrufende Anwendungssoftware als XML-Dokument zurückgeliefert.

7.3.3 Protokollierung in eRBAC: FN2RBAC-P

Die Grundfunktion Protokollierung der Informationssicherheit ist bereits Bestandteil des rollenbasierten Zugriffskontrollmodells. Die Protokollierung findet an zwei Stellen statt. Die erste Protokollierung wird direkt nach der erfolgreichen Authentifizierung einer Sitzung angelegt, in der auch die dem Subjekt zugeordneten Rollen und die zugeordneten Schlüsselattribute gespeichert werden. Die zweite Protokollierung findet bei jeder Autorisierungsanfrage statt, unabhängig davon, ob die Autorisierung verweigert oder das Subjekt autorisiert wurde.

Der Grundsatz der Zugriffskontrolle jeden Zugriffsversuch zu protokollieren kann im Konflikt mit dem Recht auf informelle Selbstbestimmung stehen. Der Datenschutz stellt hier z. T. Restriktionen in Bezug auf die Protokollierung auf.

7.4 Interaktion zwischen FN2AUTH und FN2RBAC

In diesem Abschnitt wird die Interaktion von FN2AUTH und FN2RBAC erläutert. Dabei sind wie **Abb. 7-4** zeigt, die folgenden vier Bereiche zu betrachten:

- FN2XML-Server: Ansprechbar als Webservice,
- Authentifizierungsportal (FN2AUTH) für die Authentifizierung,
- Zugriffskontrollsystem (FN2RBAC), das die Rechteverwaltung (FB2RBAC-V), Rechteprüfung (FB2RBAC-RP) und die Protokollierung (FB2RBAC-P) übernimmt und

- die FN2Klienten, die nach erfolgreicher Authentifizierung und Zugriffskontrolle für ein Objekt des Typs „Klasse“ direkt mit dem FN2XML-Server kommunizieren.

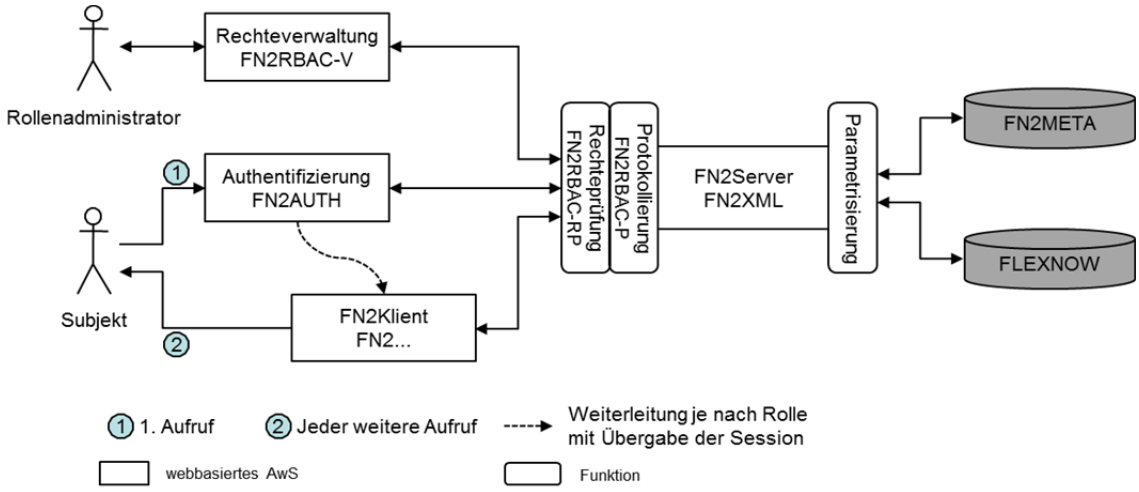

Abb. 7-4 Interaktion Authentifizierung und Zugriffskontrolle

Nach erfolgreicher Authentifizierung (siehe Kapitel 7.2) werden anhand der zugeordneten Rollen die autorisierten Anwendungen für das Subjekt zur Verfügung gestellt. Anschließend kommuniziert der jeweilige FN2Klient direkt mit dem FN2XML-Server. Nach der Rechteprüfung, die den Zugriff autorisiert und protokolliert, werden die entsprechenden Informationen vom FN2XMLServer ausgeliefert. Dabei wird falls notwendig die Domäne der Ausgabe durch die Parametrisierung eingeschränkt.

7.5 Zusammenfassung

Der in diesem Kapitel vorgestellte Zugriffskontrollmechanismus FN2RBAC ist eine exemplarische Realisierung des in dieser Arbeit entwickelten Zugriffskontrollmodells eRBAC. In FN2RBAC können die für die Zugriffskontrolle notwendigen Entitäten wie Objekttypen, Objekte, Operatoren, Zugriffsrechte, die entsprechenden Rollentypen und ihre Rollen sowie die Rollenhierarchie hinterlegt werden. Ebenfalls können die Personalisierung und die Parameter für die entsprechenden Rollen von der Administration zugeordnet werden. Durch die Objekttypen kann zusammen mit der Personalisierung eine flexible Aufrufstruktur realisiert werden. Mit der Möglichkeit der Parametrisierung kann die Anzahl der Rollen klein gehalten werden. Die Rechteprüfung selbst wird als Referenzmonitor zur Verfügung gestellt.

FN2RBAC muss die Prinzipien sicherer Zugriffskontrollsysteme für die Implementierung (siehe Kapitel 2.4.3) einhalten. Durch die Implementierung als Referenzmonitor (siehe Kapitel 2.4.4.1) kann überprüft werden, ob die Informationssicherheit gewährleistet wird. Das Prinzip des minimalen Zugriffsrechts wird erreicht, indem immer nur eine Rolle aktiviert werden kann. Durch die Implementierung eines Web-Services können auch andere Anwendungen als das Prüfungsverwaltungs-system FlexNow über FN2RBAC autorisiert werden. Die Zugriffsentscheidung einschließlich Domänenbeschränkung findet nach dem Erlaubnisprinzip statt. Durch die Überprüfung jeder von außen aufrufbaren Funktion des Anwendungssystems durch das Zugriffskontrollsystem wird die Vollständigkeit der Zugriffskontrolle sichergestellt. Das Prinzip des offenen Entwurfes wird durch die ausführliche Beschreibung des Mechanismus und Verfahrens von eRBAC gewährleistet. Die Zugriffskontrolle erfolgt für Aufgabenträger transparent und wird immer automatisch ausgeführt. Sie unterstützt damit die Akzeptanz von FN2RBAC beim Endnutzer. Die Rechteprüfung ist in genau einer Klasse im FN2XML-Server implementiert. Damit ist der fachliche Programmcode nicht mit Autorisierungscode durchsetzt und die Zugriffskontrolle findet auch für den Entwickler transparent statt. Das Prinzip der Isolierung wird gewährleistet, da nur eine Klasse die Autorisierung vornimmt und auch der FN2XML-Server, auf dem die Zugriffskontrolle stattfindet, nicht von außen erreichbar ist.

8 Fallstudie: Modellierung von Rollen mit eRBAC

In diesem Kapitel wird als Anwendungsfall anhand des Universitätsprozesses „Prüfung“ die Zugriffskontrollstrategie sowie eine konkrete Modellierung der aufgabenbezogenen Rollen einschließlich Personalisierung und Domainenbeschränkung, der Rollenhierarchie und der Zugriffsrechte aus der Aufgabenebene des IS heraus entwickelt.

In Kapitel 5 wurde herausgearbeitet, dass die Rolle als virtueller Aufgabenträger betrachtet und aus der Aufgabenebene der IS heraus entwickelt werden kann. Sie kann am Übergang zwischen der Aufgabenebene des IS und der Aufgabenträgerebene angesiedelt werden.

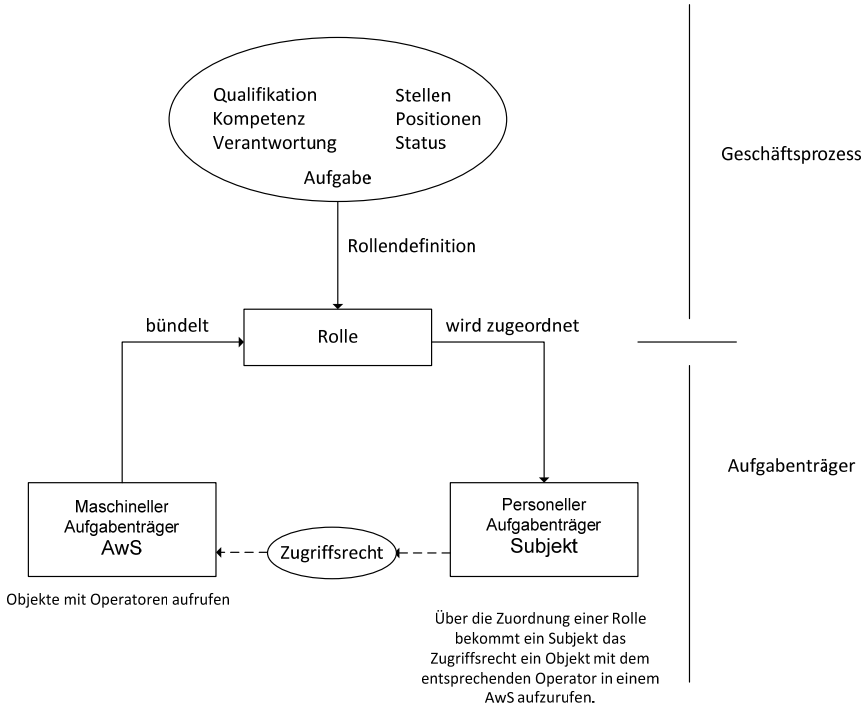

Abb. 8-1 Zusammenhang zwischen Rolle, Aufgabe und Aufgabenträger

In **Abb. 8-1** wird aus Sicht der Aufgaben- und der Aufgabenträgerebene der Zusammenhang zwischen Rolle, Aufgabe und Aufgabenträger nochmals verdeutlicht. Eine Rolle wird anhand der Aufgaben definiert. Dabei müssen Stellen, Positionen und Status innerhalb der Organisation, die dazu notwendige Qualifikation und Kompetenz sowie die übertragene Verantwortung berücksichtigt werden. Die Modellierung der aufgabenbezogenen Rollen erfolgt beim Entwurf der Zugriffskontrolle. Die Rolle befindet sich am Übergang zwischen dem Geschäftsprozess und der Zuordnung von Aufgaben zu Aufgabenträgern. Dadurch werden sowohl die Aufbauorganisation als

auch die Ablauforganisation berücksichtigt. Zur Laufzeit wirkt bei der Zugriffskontrolle auf Aufgabenträgerebene bei der Kommunikation zwischen einem personellen und maschinellen Aufgabenträger das Zugriffsrecht.

Die konkreten Rollen für eine Organisation werden in einem Role Engineering Prozess als Top-Down Ansatz ermittelt (Neumann und Strembeck 2002, S. 33). Als Methodik für die Beschreibung des Prüfungsprozesses an einer Universität wird das Semantische Objektmodell (SOM) (Ferstl und Sinz 1995; Ferstl und Sinz 2013, S. 194) verwendet. Ausgehend vom Geschäftsprozess werden dabei die Zugriffskontrollstrategie und das Zugriffskontrollmodell (Kapitel 2.4) sowie die aufgabenbezogenen Rollen festgelegt.

8.1 Geschäftsprozess der Prüfungsverwaltung

Ausgangpunkt ist der Geschäftsprozess der Prüfungsverwaltung an einer Universität. Bei der anschließenden Beschreibung wird von den Lenkungsaufgaben des Staatministeriums, den spezifischen Unterschieden für die Überwachung der Prüfungen für das Lehramt, von Prozessen der Entwicklung neuer Prüfungsordnungen (PO) und Modulhandbüchern[65] an Hochschulen abstrahiert.

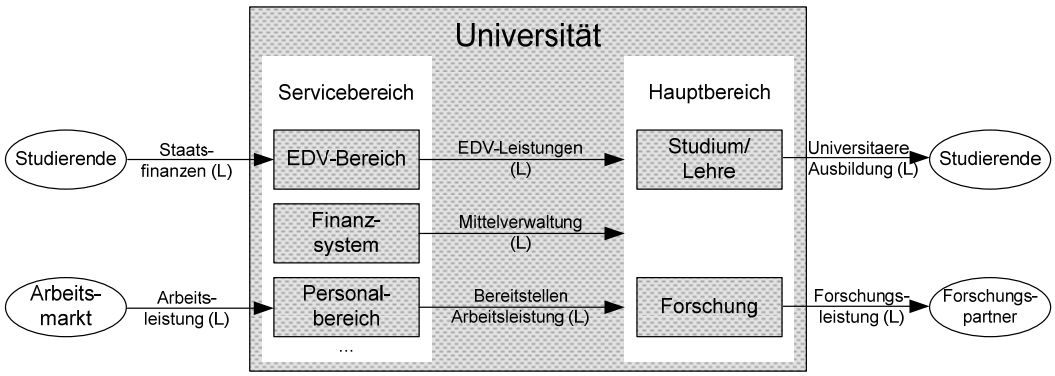

Abb. 8-2 Hauptprozesse und ausgewählte Serviceprozesse der Universität (Leistungssicht) (Sinz 1998b, S. 16)

Der Hauptbereich einer Universität kann in die zwei Geschäftsprozesse: Studium und Lehre sowie Forschung zerlegt werden. Diese beiden Prozesse werden u. a. unterstützt von den Prozessen EDV-Bereich, Finanzsystem und Personalbereich des Servicebereiches, siehe **Abb. 8-2.**

[65] Die in einer Prüfungsordnung genannten Module werden im dazugehörigen Modulhandbuch konkretisiert.

Der Prozess „Studium und Lehre“ beinhaltet den Teilprozess „Prüfung“, dessen Sachziel der Nachweis der Leistungsfähigkeit des einzelnen Studierenden ist. Prüfungen sind Teil der erbrachten Leistung **Universitäre Ausbildung** (Sinz und Krumbiegel 1996; Sinz 1998a, S. 6; Sinz 1998b, S. 5).

Studierende sind Empfänger der Leistung. Gemäß der SOM-Methodik wird diese Leistung

- durch Veröffentlichung eines Prüfungsangebots angebahnt (Anbahnungsphase),
- durch Anmeldung und Zulassung zu Prüfungen vereinbart (Vereinbarungsphase) und
- anschließend im Rahmen der Prüfungsdurchführung erbracht (Durchführungsphase) (Sinz 1998b, S. 17).

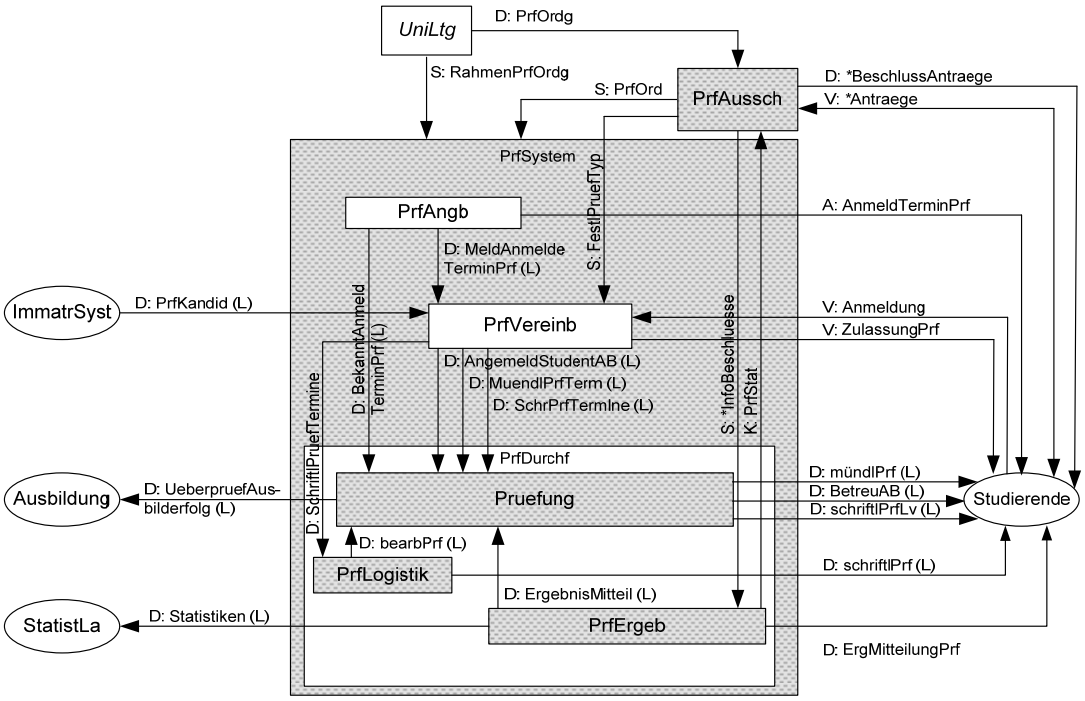

Abb. 8-3 Ausschnitt aus der Leistungssicht des Prüfungssystems (Sinz und Krumbiegel 1996)

Das Objekt **Prüfungsausschuss** (**PrfAussch**) (siehe **Abb. 8-3)** ist das verantwortliche Gremium und achtet darauf, dass die Bestimmungen der Prüfungsordnung eingehalten werden. Es sorgt im Benehmen mit dem Prüfungsamt für eine ordnungsgemäße Durchführung der Prüfungen. Es steuert durch die Prüfungsordnung das Prüfungssystem (**PrfSystem**). Die Universitätsleitung (**UniLtg**) gibt nach Genehmigung der Prüfungsordnung (**PrfOrd**) diese an den Prüfungsausschuss wei-

ter. Die Prüfungsordnung hat damit Einfluss auf die Objekte Prüfung (**Pruefung**) und Prüfungsvereinbarung (**PrfVereinb**). Der Prüfungstyp kann zu Beginn des Semesters vom Prüfungsausschuss bzw. Prüfer festgelegt werden und wird dem Objekt Prüfungsvereinbarung mitgeteilt (**FestlPruefTyp**) (Sinz und Krumbiegel 2006).

Auch das Ausnahmeverhalten wird über das Objekt Prüfungsausschuss abgewickelt. Ein Studierender kann Anträge (***Antraege**)[66] zur Anerkennung von Prüfungsleistungen oder aufgrund von Mängeln im Prüfungsverfahren z.B. einen Antrag auf Rücktritt von der Prüfung stellen. Über diese Anträge entscheidet der Prüfungsausschuss und sendet einen Beschluss (***BeschlussAntraege**) zurück. Das Objekt Prüfungsergebnis wird ebenfalls über die Beschlüsse des Prüfungsausschusses informiert (***InfoBeschluesse**) (Sinz und Krumbiegel 2006).

Das Prüfungssystem (**PrfSystem**) wird von dem Steuerobjekt Universitätsleitung (**UniLtg**) durch die Rahmenprüfungsordnung (**RahmenPrfOrdg**) und vom Prüfungsausschuss durch die Prüfungsordnung (**PrfOrdg**) gesteuert (Sinz und Krumbiegel 2006). Innerhalb des Prüfungssystems erbringen folgende Objekte (siehe **Abb. 8-3)** die notwendigen Serviceleistungen:

Das Objekt **Prüfungsangebot (PrfAngb)** wickelt die Anbahnungsphase von Prüfungen ab und übernimmt dabei die Bekanntgabe der An- bzw. Abmeldetermine zu Prüfungen für Studierende und meldet diese Termine dem Objekt **Prüfungsvereinbarung** und gibt diese ebenso dem Objekt **Prüfung** bekannt.

- Das Objekt **Prüfungsvereinbarung** (**PrfVereinb**) übernimmt die Abwicklung der Anmeldungen zu Prüfungen. Dabei ist das Objekt zuständig für die Anmeldung der Studierenden zu den Prüfungen einschließlich Abschlussarbeit, Zulassung zu Prüfungen sowie Koordination und Veröffentlichung der Termine für schriftliche und mündliche Prüfungen. Teilweise werden Prüfungstermine zwischen dem Umweltobjekt **Studierenden** und dem Objekt **Prüfung** direkt vereinbart (**MuendlPrfTermin**). Vom Objekt **Immatrikulationssystem** (**ImmatrSyst**) werden dazu die **Prüfungskandidaten** (**PrfKandid**) bezogen.

[66] „*“ kennzeichnet die Prozesse des Ausnahmeverhaltens.

- Das Objekt **Prüfungsdurchführung (PrfDurchf)** besteht aus den Objekten: Prüfung (**Pruefung**), Prüfungslogistik (**PrfLogistik**) und Prüfungsergebnis (**PrfErgeb**).
- Das Objekt **Pruefung** übernimmt für das Objekt **Ausbildung** die Überprüfung des Ausbildungserfolges. Es erbringt die Hauptleistungen: mündliche Prüfung, schriftliche Prüfung einer lehrveranstaltungsgebundenen Prüfung (**schriftlPrfLv**), Betreuung der Abschlussarbeit sowie über das Objekt **Prüfungslogistik** für zentral organisierte Prüfungen die Leistung schriftliche Prüfung (**schriftlPrf**). Für lehrveranstaltungsgebundene Prüfungen wird die gesamte organisatorische Abwicklung übernommen.
- Das Objekt **Prüfungslogistik** übernimmt die organisatorische Abwicklung der zentral organisierten schriftlichen Prüfungen (**schriftlPrf**).
- Das Objekt **Prüfungsergebnis** übernimmt für das Objekt **Prüfung** die Bekanntgabe der Prüfungsergebnisse, das Ausstellen von Bescheinigungen und die Gewährung der Einsicht in die Prüfungsunterlagen für Studierende (**ErgebnisMitteil**). Außerdem erstellt es Statistiken (**Statistiken**) für das Statistische Landesamt (**StatistLa**) und Prüfungsstatistiken (**PrfStat**) für den Prüfungsausschuss.

In **Tab. 8-1** sind den jeweiligen Objekten die dazugehörigen Aufgabenträger bzw. Organisationseinheiten zugeordnet, die die Aufgaben dieses Objektes durchführen.

Tab. 8-1 Objekte und Organisationseinheiten des Prüfungsprozesses

Objekt	Aufgabenträger/Organisationseinheit
Prüfung	Prüfer
Prüfungsangebot (PrfAngb)	Prüfungsamt der Universität, Prüfer
Prüfungsausschuss (PrfAussch)	Prüfungsausschuss
Prüfungsergebnis (PrfErgeb)	Prüfungsamt der Universität, Prüfer
Prüfungslogistik (PrfLogistik)	Prüfungsamt der Universität, Prüfer
Prüfungsvereinbarung (PrfVereinb)	Prüfungsamt der Universität, Prüfer

8.2 Zugriffskontrollstrategie

Nach der Beschreibung der einzelnen Schritte des Prüfungsprozesses gilt es, aus den drei in Kapitel 2.4.1 vorgestellten Zugriffskontrollstrategien, die für eine Prüfungsverwaltung geeignete Strategie auszuwählen:

- Benutzerbestimmte Zugriffskontrollstrategie
- Systemweite Zugriffskontrollstrategie
- Rollenbasierte Zugriffskontrollstrategie

Ein Prüfungsverwaltungssystem speichert Informationen wie persönliche Daten und Prüfungsverläufe von Studierenden. Ein Mitarbeiter des Prüfungsamtes, Prüfender oder Studierender darf nicht entscheiden, wer auf welche Daten Zugriff erhält. Eine benutzerbestimmte Zugriffskontrollstrategie scheidet damit aus, da ein einzelner Aufgabenträger im Sinne der Zugriffskontrolle kein Eigentümer oder Besitzer (siehe Kapitel 2.4.1) von Prüfungsleistungen ist. Aus der Untersuchung der Aufgaben ergibt sich, dass die gleichen Objekte nacheinander von verschiedenen Aufgabenträgern als Subjekt bearbeitet oder geändert werden müssen.

Eine systemweite bzw. hochschulweite Zugriffskontrollstrategie ist für das Durchsetzen einer einheitlichen Sicherheitsstrategie, wie sie für die Umsetzung des Prüfungsprozesses notwendig ist, durchaus geeignet. Es lassen sich jedoch aus dem Prüfungsprozess keine sinnvollen hierarchischen Sicherheitskategorien ableiten. Das Recht eines Studierenden, eine Prüfung anzumelden, ist nicht niedriger einzustufen als die Anmeldung einer Prüfung durch einen Mitarbeiter im Prüfungsamt. Während der Anmeldung zu einer Prüfung in der Vereinbarungsphase ist es sogar erwünscht, dass der Studierende seine Daten verändert, indem er sich selbst zu einer Prüfung anmeldet. Dadurch wird eine Entlastung des personellen Aufgabenträgers im Prüfungsamt erreicht.

Wird der Prüfungsprozess näher betrachtet, so müssen innerhalb der einzelnen Phasen Aufgaben erbracht werden, deren Nachereignisse die Vorereignisse für die nachfolgende Phase bilden. Diese Aufgaben werden durch die zugeordneten personellen Aufgabenträger ausgeführt. Die Rolle bündelt die benötigten Zugriffsrechte und nimmt für den jeweiligen Aufgabenträger die Zugriffskontrolle vor, damit dieser seine teilautomatisierten Aufgaben erledigen kann. Deshalb kommt für die Umsetzung der Zugriffskontrolle nur eine rollenbasierte Zugriffskontrollstrategie in Frage. Dadurch ist gleichzeitig das Zugriffskontrollmodell festgelegt. Wie zudem in Kapitel 3 herausgearbeitet, ist das rollenbasierte Zugriffskontrollmodell am besten geeignet, eine rollenbasierte Zugriffskontrollstrategie umzusetzen. Das entwickelte eRBAC (siehe Kapitel 6) stellt die Entitäten zur Verfügung, um die Autorisierung für ein Anwendungssystem modellieren und in einem Zugriffskontrollsystem umsetzen zu können.

8.3 Aufgabenbezogene Rollen im Prüfungsprozess an Hochschulen

Dieses Kapitel dokumentiert eine Modellierung der konkreten Rollen in einem Top-Down-Ansatz anhand der Aufgaben des Prüfungsprozesses an Hochschulen mit Hilfe des Zugriffskontrollmodells eRBAC. Ausgehend vom Prüfungsprozess werden die Zugriffsrechte, die Rollen und die Rollenhierarchie modelliert. Die Identifikation und Konzeption von Rollen und Rollenhierarchien ist ein komplexer Vorgang. Um Rollen zu finden, wird Wissen über die beteiligten Anwendungssysteme, die Administration der beteiligten Plattformen als auch ein detailliertes Wissen über die Aufgaben der jeweiligen Geschäftsprozesse benötigt (Roeckle et al. 2000, S. 103).

8.3.1 Aufgaben und Aufgabenträger im Prüfungsprozess

Folgende Aufgabengrobgliederung des Geschäftsprozesses Prüfungsverwaltung kann nach Phasen aus Kapitel 8.1 abgeleitet werden.
Vor der Anbahnungsphase sind folgende vorbereitende Aufgaben notwendig:

- Stammdatenübernahme aus dem Immatrikulationssystem (ImmatrSyst)
- Abbildung der Regeln einer Prüfungsordnung im AwS

In der Anbahnungsphase sind folgende Aufgaben notwendig:

- Erstellen des Prüfungsangebotes für den jeweiligen Prüfungszyklus
- Bekanntgabe der Anmeldetermine an das Umweltobjekt Studierende

In der Vereinbarungsphase erfolgen zwischen den Objekten **PrfVereinb** und dem Umweltobjekt **Studierende** die:

- Anmeldungen zu Prüfungen
- Abmeldungen von Prüfungen
- Zulassung zur Prüfung bei Kapazitätsbeschränkungen oder Abschlussarbeiten
- Setzen von Freiversuchen

Die Prüfungsdurchführung erfolgt nach der An- und Abmeldephase:

- Raum- und Zeitplanung der Prüfungen
- Bekanntgabe der Prüfungstermine und Räume der Prüfungen
- Erfassung der Korrekturergebnisse durch die Lehrstühle
- Verbuchen der ECTS-Punkte und Veröffentlichung der Ergebnisse

- Erstellen von Mitteilungen und Bescheiden
- Erstellen von Kontoauszügen für Studierende
- Ergebnisrechnung und Druck der Ergebnisse

Diese Aufgaben lassen sich Rollen zuordnen, die die notwendigen Zugriffsrechte besitzen, um diese erledigen zu können. Die Rolle **Studierender** kann u. a. folgende Aufgaben wahrzunehmen:

- Lehrveranstaltungen anmelden,
- Lehrveranstaltungen abmelden,
- Prüfungen anmelden,
- Prüfungen abmelden,
- Ergebnisse einsehen,
- Freiversuche setzen,
- Leistungsnachweis, Kontoauszüge ausdrucken,
- TAN-Liste[67] ausdrucken[68],
- Einstellungen vornehmen z. B. E-Mail-Versand der Ergebnisse erlauben.

Die Aufgabenträger, die am Prüfungsprozess beteiligt sind, lassen sich grob wie folgt unterteilen (Sinz und Krumbiegel 1996):

- Modellierer[69] (Abbildung der Regeln einer Prüfungsordnung),
- Prüfungsamt (Anbahnung für zentral organisierte Prüfungen, Prüfungslogistik und Prüfungsdurchführung),
- Lehrstühle bzw. Institute (Anbahnung und Vereinbarung für lehrveranstaltungsgebundene Prüfungen sowie die Prüfungsdurchführung der lehrveranstaltungsgebundenen und zentral organisierten Prüfungen),
- Studierende (Vereinbarung, Ergebnisse einsehen, Leistungsnachweise ausdrucken).

[67] TAN ist die Abkürzung von Transaktionsnummer Metzger und Siller (2014)
[68] TANs autorisieren als Einmalpasswort in FlexNow An- bzw. Abmeldung von Prüfungen
[69] Die Rolle Modellierer wird im Weiteren nicht weiter untersucht, da hier alle Aufgaben in einem AwS in ihrer Gesamtheit zur Verfügung stehen.

8.3.2 Modellierung der Rollen

Aus dem im vorherigen Kapitel beschriebenen Prüfungsprozess und den exemplarisch aufgeführten Aufgaben sowie der Grobgliederung der Aufgabenträger ergeben sich folgende vier Hauptrollen (**Abb. 8-3** und **Tab. 8-1**):

- Studierende als Empfänger der Leistung Prüfung
- Prüfender als Lieferant der Leistung Prüfung
- Prüfungsausschussvorsitzende als Überwacher des Prüfungsprozesses
- Prüfungsamt als Servicelieferant für die Feststellung des Ausbildungserfolges

Das Prüfungsamt ist der Aufgabenträger für die vier Objekte Prüfungsangebot, Prüfungsvereinbarung, Prüfungslogistik und Prüfungsergebnis (siehe **Tab. 8-1**). Die Objekte Prüfungsangebot und Prüfungslogistik erbringen die Serviceleistung in den Fällen, in denen die Prüfungen zentral organisiert werden. Im Folgenden werden die Leistungs- und Lenkungsaufgaben, die die einzelnen Objekte erbringen, näher betrachtet und dahingehend gekennzeichnet, ob die Leistungen teilautomatisierbar, automatisierbar oder nicht-automatisierbar sind.

Bei lehrveranstaltungsgebundenen Prüfungen übernimmt der Aufgabenträger Prüfender selbst die Aufgaben der Objekte Prüfungsangebot und Prüfungslogistik.

> *Beispiele für lehrveranstaltungsgebundene Prüfungen sind Seminare mit der Prüfungsleistung Hausarbeit oder Projektarbeit.*

Das Prüfungsangebot übernimmt folgende Aufgaben:

- Für die Bekanntgabe der Anmeldefristen werden intern Prüfungsangebote für zentral organisierte Prüfungen im jeweiligen Semester erstellt (teilautomatisierbar).
- Bekanntgabe der Anmeldefristen für Prüfungen (**BekanntAnmeldTerminPrf**) an das Objekt Prüfung und Meldung der Anmeldetermine zu Prüfungen (**MeldAnmeldeTerminPrf**) an das Objekt Prüfungsvereinbarung sowie Weiterleitung der Termine an das Umweltobjekt Studierende (**AnmeldTerminPrf**) (automatisierbar).

Die Prüfungsvereinbarung übernimmt folgende Aufgaben:

- Bekanntgabe der Termine für schriftliche und mündliche Prüfungen an das Objekt Prüfungslogistik (**SchriftlPruefTermine**) (automatisierbar).

- Der Studierende meldet sich über die Prüfungsvereinbarung für eine Prüfung an (**Anmeldung**). Dabei wird zwischen den Anmeldungen zu schriftlichen und mündlichen[70] Prüfungen und Abschlussarbeit unterschieden.
    - Anmeldung zu schriftlichen und mündlichen Prüfungen sowie Hausarbeiten (teilautomatisierbar).
    - Anmeldung zu Abschlussarbeiten (teilautomatisierbar).
- Nach der Prüfungsanmeldung wird der Studierende zur Prüfung zugelassen (**ZulassungPrf**) (teilautomatisierbar).
- Weitergabe aller angemeldeten Studierenden bestehend aus den drei Serviceleistungen: **SchrPrfTermine**, **AngemeldStudentAB** und **MuendlPrfTerm** (automatisierbar).

Das Objekt Prüfungslogistik übernimmt folgende Aufgaben:

- Organisatorische Abwicklung also Planung und Durchführung der zentral organisierten schriftlichen Prüfungen für das Objekt Prüfung (teilautomatisierbar).
- Austeilen der Prüfungen an den Studierenden (**schriftPrf**) (nicht automatisierbar).
- Weiterleitung der zu bearbeitenden Prüfungen an das Objekt Prüfung (**bearbPrf**) (nicht automatisierbar).

Das Objekt Prüfungsergebnis übernimmt folgende Aufgaben:

- Verteilung von Prüfungsprotokollen für die Durchführung der mündlichen Prüfung (**mündlPrf**) (teilautomatisierbar).
- Mitteilung der Prüfungsergebnisse an das Umweltobjekt Studierende (**ErgMitteilungPrf**) (teilautomatisierbar).
- Bearbeiten von Ausnahmeverhalten nach Mitteilung durch den Prüfungsausschuss (***InfoBeschluesse**) (teilautomatisierbar).
- Erstellen von Statistiken für den Prüfungsausschuss (**PrfStat**) (automatisierbar).
- Erstellen von Statistiken für das Statistische Landesamt (**StatistLa**) (automatisierbar).

[70] Unter schriftlicher Prüfung sind auch Hausarbeiten oder Projektarbeiten zu verstehen. Unter mündlicher Prüfung werden auch Referate und Vorträge subsumiert.

- Bescheiderstellung und Zeugniserstellung für das Objekt Pruefung (**ErgebnisMitteil**) (teilautomatisierbar).

Die Rolle Prüfungsamt kann in vier Rollen gemäß den Objekten im Prüfungsprozess, die die Serviceleistung für das Objekt Prüfung erbringen, zerlegt werden:

- Prüfungsangebot (PrfAng)
- Prüfungsvereinbarung (PrfVereinb)
- Prüfungslogistik (PD)
- Prüfungsergebnis (PA)

Diese Rollen werden untersucht, in wie weit sie dieselben Zugriffsrechte besitzen. Damit werden aus dem Prüfungsprozess virtuellen Rollen und Anwendungsrollen (siehe **Tab. 8-2**) modelliert. Virtuellen Rollen (siehe Kapitel 6.2) werden zuerst festgelegt, da diese Zugriffsrechte, die mehrere Rollen gemeinsam haben, bündeln. Danach werden die Anwendungsrollen entwickelt und beide in eine Rollenhierarchie eingeordnet.

Tab. 8-2 Virtuelle Rollen, die in die Rollenhierarchie eingebunden werden

Rolle	Beschreibung
Nutzer	kapselt Zugriffsrechte, die jeder Aufgabenträger benötigt.
Lv.An-Abmelden	kapselt Zugriffsrechte, um An- bzw. Abmeldung einer Lehrveranstaltung vorzunehmen.
LvPrf.An-Abmelden	kapselt Zugriffsrechte, um eine An- bzw. Abmeldung einer lehrveranstaltungsgebundenen Prüfung vorzunehmen.
PrfZentral.An-Abmelden	Zugriffsrechte für eine An- bzw. Abmeldung einer zentral organisierten Prüfung.
Ergebnisse.Einsehen	Erlaubnis: Ergebnisse von Prüfungen einzusehen. Diese virtuelle Rolle ist eine parametrisierte Rolle (siehe Kapitel 8.3.7).

Es existieren an Hochschulungen Anwendungsfälle, bei denen die Rolle LM nochmals unterteilt wird, weil dort Aufgaben stärker dezentralisiert sind. Deshalb ist eine direkte Zuordnung dieser Rollen zu Aufgabenträgern notwendig und dementsprechend werden die Rollen Katalog.Verwalten, Lv.Verwalten, PrfZentral.Noteneingeben und LvPrf.Noteneingeben nicht als virtuelle Rollen modelliert.

Tab. 8-3 Rollen, die Subjekten zugeordnet werden können

Rolle	Beschreibung
Katalog.Verwalten	Eingabe aller Grunddaten, die für einen Katalog[71] notwendig sind.
Lv.Verwalten	Lehrveranstaltungen anlegen und verwalten.
PrfZentral.Noteneingeben	Prüfungsergebnisse für zentral organisierte Prüfungen erfassen.
LvPrf.Noteneingeben	Prüfungsergebnisse für lehrveranstaltungsgebundene Prüfungen erfassen.

[71] Ein Katalogeintrag beschreibt die Typinformationen einer Lehrveranstaltung. Von diesem Katalogeintrag können jedes Semester beliebig viele Instanzen erzeugt werden.

LM	Diese Rolle umfasst alle Aufgaben eines Prüfers.
PrfAng	Alle Aufgaben, die sich mit der Erstellung und Veröffentlichung des Prüfungsangebotes für zentral organisierte Prüfungen befassen.
PrfVereinb	Alle Aufgaben, die mit der Prüfungsvereinbarung von zentral organisierten Prüfungen notwendig sind.
PD	Alle Aufgaben, die für die Prüfungslogistik erforderlich sind.
PA	Alle Aufgaben, die zur Bearbeitung von Ausnahmeregelungen bei Prüfungen notwendig sind.
PAVOR	Kapselt alle Zugriffsrechte für die Aufgaben eines Prüfungsausschussvorsitzenden, um die notwendigen Informationen zu erhalten.
Studierender	Kapselt Zugriffsrechte für Serviceaufgaben wie TAN-Liste ausdrucken und Einstellungen festlegen.

8.3.3 Aufgabentrennung in der Prüfungsverwaltung

In dem Geschäftsprozess „Prüfung“ werden die beiden Aspekte der Aufgabentrennung benötigt (Kapitel 2.4.3, 4.2.1): Der funktionale Aspekt der Aufgabentrennung existiert z. B. zwischen Studierenden und Mitarbeitern im Prüfungsamt. Ist einem Subjekt die Rolle **Studierenden** zugeordnet, so darf diesem nicht auch noch die Rolle **PA** zugeordnet werden. Der funktionale Aspekt der Aufgabentrennung zwischen den Rollen **PA** und **Student** wird als statische Aufgabentrennung umgesetzt.

Das 4-Augenprinzip ist zwischen der Rolle **LM** und der Rolle **PA** zu finden. Die Noten werden direkt von Prüfern eingetragen. Nach erfolgter Noteneingabe wird eine Mitteilung an das Prüfungsamt gesendet und im Prüfungsamt erfolgt die Freigabe und Verbuchung der Noten. In diesem Fall wird zwischen den Rollen **LM** und **PA** eine dynamische Aufgabentrennung modelliert: d.h. diese beiden Rollen dürfen nicht in derselben Sitzung aktiviert werden. Es muss zuerst die aktive Rolle verlassen werden, bevor die andere Rolle aktiviert werden kann.

8.3.4 Rollenhierarchie im Prüfungsprozess

Aus den in Kapitel 8.3.2 ermittelten Rollen kann die in **Abb. 8-4** dargestellte Rollenhierarchie gebildet werden. Von den einzelnen Zugriffsrechten der Rollen wird in der Abbildung abstrahiert. Zu beachten ist bei der Bildung der Hierarchie, dass es sich um eine Zugriffsrechtsvererbung handelt, d. h. diese Rollen bündeln Zugriffsrechte, um Aufgaben durchführen zu können (siehe Kapitel 4.3 und 6.4).

Eine Aufgabe, z. B. Anmeldungen zu Lehrveranstaltungen, kann nicht nur von der Rolle **Studierender**, sondern auch von der Rolle **LM** ausgeführt werden. Deshalb wird eine virtuelle Rolle **Lv: An- bzw. Abmelden** gebildet und beiden Rollen zuge-

ordnet. Die Rolle **PrfAng**, **PrfVereinb** und **PD** wird in der Hierarchie nicht dargestellt, da diese nur von der virtuellen Rolle **Nutzer** und nicht von weiteren Rollen erben. Die gebildete Hierarchie sagt nichts über die Aktivierung der Rollen aus. Dafür ist eine getrennte Rollenhierarchie, die Aktivierungshierarchie zu bilden.

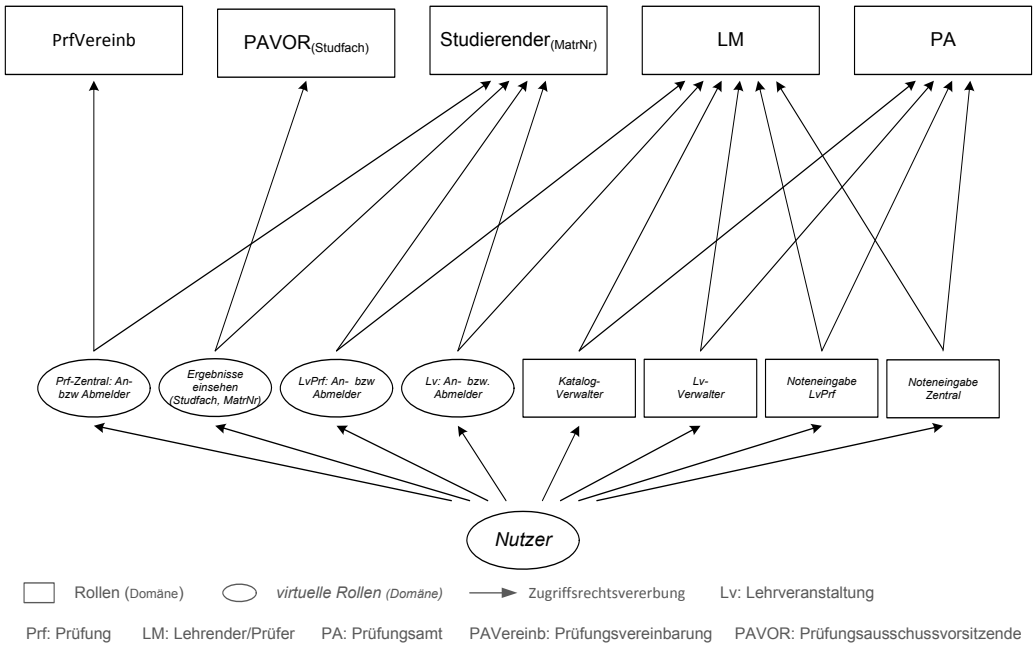

Abb. 8-4 Rollenhierarchie für FN2RBAC für ein Prüfungsverwaltungssystem

Im Anschluss an die Rollendefinition werden die teilautomatisierten Aufgaben dahingehend untersucht, welche Funktionen im AwS dazu benötigt werden. Für jede von außen aufrufbare Funktion werden entsprechende Objekte, Operatoren und Zugriffsrechte im Zugriffskontrollsystem benötigt.

8.3.5 Modellierung der Zugriffsrechte

In diesem Abschnitt werden ausgewählte Aufgaben für die Zugriffskontrolle in Funktionen zerlegt, Zugriffsrechte mit Objekten und Operatoren ermittelt und den entsprechenden aufgabenbezogenen Rollen zugeordnet:

Die Aufgabe **Prüfung (lehrveranstaltungsgebunden) anmelden** kann von drei Rollen ausgeführt werden: Studierenden, LM und PA. Ein Studierender meldet sich normalerweise selbstständig zu seinen Prüfungen an. Der Lehrstuhl kann bei Bedarf die Anmeldung ebenfalls vornehmen. Sind vor der Anmeldung noch besondere Voraussetzungen von Seiten des Prüfungsamtes zu prüfen, erfolgt die Anmeldung im Prüfungsamt. Je nach Rolle sind unterschiedliche Prozessschritte notwendig, um alle Informationen zu erhalten und die Anmeldung durchführen zu können. Anschließend

werden exemplarisch Zugriffsrechte pro Rolle beschrieben, die notwendig sind, um die Anmeldung durchzuführen.

Für das Anmelden einer Prüfung in der Rolle Studierender sind u.a. folgende Zugriffsrechte erforderlich:

- Zugangsdaten eingeben und damit implizit Sitzung erzeugen
- Anwendung FN2STUD aufrufen
- Anmeldbare Module holen
- Modul auswählen
- Konkrete Teilprüfung auswählen
- Prüfungsangebot auswählen
- Ggf. Prüfer auswählen
- Prüfungsanmeldung schreiben

Für das Anmelden einer Prüfung in der Rolle LM sind u.a. folgende Zugriffsrechte notwendig:

- Zugangsdaten eingeben und damit implizit Sitzung erzeugen
- Anwendung FN2LM aufrufen
- Semesterliste holen, um das Semester für die Anmeldung auswählen zu können
- Lehrveranstaltung auswählen, in der die Prüfung erfolgen soll
- Prüfungsangebot holen
- konkrete Teilprüfung des Studierenden auswählen
- Prüfungsanmeldung schreiben

Für das Anmelden einer Prüfung in der Rolle PA sind u. a. folgende Zugriffsrechte notwendig:

- Zugangsdaten eingeben und damit implizit Sitzung erzeugen
- Anwendung FN2PA aufrufen
- Student suchen
- Studienverlauf holen, damit die Prüfung an der richtigen Stelle eingefügt werden kann
- Semesterliste holen, um das Semester für die Anmeldung auswählen zu können

- Teilprüfung auswählen
- Prüfungsangebot auswählen
- Prüfungsanmeldung schreiben

Virtuelle Rollen

Aus der Liste der Zugriffsrechte ist ersichtlich, dass alle drei Rollen das Zugriffsrecht **Sitzung erzeugen** benötigen. Hierfür wird folgende virtuelle Rolle erzeugt: **Nutzer -** diese kapselt alle Zugriffsrechte, die jeder personelle Aufgabenträger benötigt. Über die Rollenhierarchie stehen diese Zugriffsrechte den Subjekten zur Verfügung. Das Zugriffsrecht **Prüfungsanmeldung schreiben** benötigen diese drei Rollen ebenfalls, aus diesem Grund wird z. B. die virtuelle Rolle **LvPrf.An- und Abmelden** gebildet, die alle Zugriffsrechte umfasst, die von diesen drei Rollen benötigt werden.

Objekt und Operator

Zugriffsrechte werden aus Objekt und Operator gebildet. Aus den obigen Zugriffsrechten lassen sich folgende Objekte erkennen: Sitzung, Anwendung, Studierender, Studienverlauf, Teilprüfung, Prüfungsanmeldung, Modul, Semester und Prüfungsangebot.

Tab. 8-4 Beschreibung ausgewählter Objekte

Objekt	Beschreibung
Semester	Dieses kapselt alle Informationen eines Semesters.
Teilprüfung	Teilprüfung ist eine konkrete Prüfung, zu der ein Studierender angemeldet werden kann.
Prüfungsanmeldung	Das Objekt Prüfungsanmeldung entspricht einer Klasse, in der alle Methoden zusammenlaufen, die für die Anmeldung einer Prüfung notwendig sind. Diese Klasse kapselt die Informationen über die Modellierung der Prüfungsordnung bis hin zum individuellen Studienverlauf eines Studierenden.

Alle von außen aufrufbaren Objekte müssen für die Rechteprüfung modelliert werden. Das Zugriffskontrollmodell eRBAC ermöglicht diese anwendungsspezifisch zu modellieren. Objekte können dabei unterschiedliche Mengen an Information kapseln. Beispielhaft sind hier zu nennen: Teilprüfung, Datenblatt, Fakultät. **Teilprüfung** ist ein Eintrag einer konkreten Prüfung für einen Studierenden. **Datenblatt** ist der gesamte Prüfungsverlauf eines Studierenden. **Fakultät** sind die Abbildung der Daten der Fakultät.

Die Operatoren eines Anwendungssystems sind z. B. **read**, **setNote**, **getById**.

Die Zugriffsrechte in eRBAC können objektspezifisch umgesetzt werden und sind die Kombination aus den Objekten mit den dazugehörigen Operatoren. **Teilprüfung.setNote**, **Datenblatt.read**, **Fakultät.getById.** Die Zugriffsrechte beziehen sich auf die von außen aufrufbaren Methoden der Objekte.

8.3.6 Objekttypen und Personalisierung der Rollen

Objekte vom Typ **Anwendung** sind die webbasierten Anwendungen für Lehrstühle (FN2LM), für Studierende (FN2STUD) oder für Prüfungsausschussvorsitzende (FN2PA). Die dazugehörigen Zugriffsrechte lauten. FN2LM.open, FN2STUD.open und FN2PA.open. Der Rolle LM ist genau ein Objekt des Typs Anwendung zugeordnet. Die Rolle LM ist zudem eine Rolle, die durch eine Organisationseinheit personalisiert werden muss, da von Mitarbeitern der Lehrstühle nur Prüfungen bearbeitet werden dürfen, für die diese zuständig sind. Nach der eventuellen automatischen Auswahl der webbasierten Anwendung FN2LM wird nun überprüft, ob der Aufgabenträger für mehr als einen Lehrstuhl arbeitet. Ist dies der Fall, wird eine Auswahlmöglichkeit zur Verfügung gestellt, wenn nicht, wird automatisch weitergeleitet.

Alle Operatoren, die sich auf Funktionen innerhalb des Anwendungssystems beziehen, sind Objekten des Objekttyps **Klasse** zugeordnet. Jede dieser Kombination bildet ein Zugriffsrecht.

8.3.7 Domänenbeschränkung

Ein Prüfungsausschussvorsitzender darf nur Studienverläufe von Studierenden sehen, die in den jeweiligen Zuständigkeitsbereich fallen. Deshalb ist die Rolle PAVOR als parametrisierte Rolle umgesetzt. Für die Domänenbeschränkung existieren zwei mögliche Parameter: Studienfach und Fakultät. Diese werden der Rolle PAVOR zugeordnet. Wird einem Aufgabenträger die Rolle PAVOR zugeordnet, so muss nach der Subjektzuordnung für diesen Aufgabenträger, entweder die erlaubten Studienfächer oder Fakultäten, festgelegt werden. Da die Domänenbeschränkung nach dem Erlaubnisprinzip erfolgt, wird, wenn diese Zuordnung nicht vorgenommen wurde, beim Suchen von Studierenden, kein Studierender angezeigt.

8.4 Zusammenfassung

In diesem Kapitel wurden exemplarisch für den Prüfungsprozess die Rollen aus den Aufgaben des betrieblichen Informationssystems heraus entwickelt. Diese aufgabenbezogener Rollen bündeln die notwendigen Zugriffsrechte, die für Erledigung der Aufgaben eines personellen Aufgabenträgers an der Schnittstelle zu einem maschinellen Aufgabenträger notwendig sind. Es kann damit gezeigt werden, dass eRBAC alle Entitäten für Modellierung aufgabenbezogenen Rollen mit ihren Zugriffsrechten und Beschränkungen zur Verfügung stellt.

Die Durchsetzung der rollenbasierten Zugriffskontrollstrategie kann mit dem Zugriffskontrollmechanismus FN2RBAC durchgeführt werden. Basierend auf dem erweiterten Zugriffskontrollmodell eRBAC werden in der Rechteverwaltung FN2RBAC–V die Entitäten gespeichert. Die Rechteprüfung übernimmt FN2RBAC-RP und die Protokollierung wird von FN2RBAC-P durchgeführt.

9 Zusammenfassung und Ausblick

Die vorliegende Arbeit beschäftigt sich mit der Gewährleistung der betrieblichen Informationssicherheit, insbesondere IT-Sicherheit an der Kommunikationsschnittstelle zwischen personellen und maschinellen Aufgabenträgern. Durch die Untersuchung des Konzepts der Rolle im betrieblichen Informationssystem als Bündelung von Berechtigungen an Hand von Aufgaben, wird eine Lücke zwischen den beiden Forschungsgebieten Informationssicherheit und betriebliches Informationssystem geschlossen. Mit den gewonnenen Erkenntnissen wurde das Referenzmodell des rollenbasierten Zugriffskontrollmodells zu eRBAC als Grundlage eines Zugriffskontrollsystems für Anwendungssysteme erweitert.

Zusammenfassung

Im ersten Teil der Arbeit wurde der Untersuchungsrahmen abgesteckt und in die Terminologie der Informationssicherheit eingeführt. Außerdem wurde für die Auswahl eines Zugriffskontrollmodells ein Klassifikationsrahmen erarbeitet und anhand dessen ein Zugriffskontrollmodell ausgewählt.

Die zu gewährleistenden Sachziele der betrieblichen Informationssicherheit sind Vertraulichkeit, Integrität und Verbindlichkeit; wobei im betrieblichen Informationssystem Integrität an erster Stelle steht. Die Erfüllung diese Sachziele wird durch Authentifizierung und Zugriffskontrolle sichergestellt. Die Authentifizierung wird mittels Wissen, Besitz, biometrischer Verfahren oder einer Kombination dieser durchgeführt. Sie ist die notwendige Voraussetzung der Zugriffskontrolle. Ist die Authentifizierung gescheitert, ist auch die Zugriffskontrolle als gescheitert anzusehen.

Aus der Sicherheitsstrategie des Unternehmens wird die Zugriffskontrollstrategie abgeleitet und muss durch das gewählte Zugriffskontrollmodell unterstützt werden. Die Zugriffskontrolle wird mit den Grundfunktionen Rechteverwaltung, Rechteprüfung und Protokollierung realisiert. Das Zugriffskontrollmodell bildet die Grundlage für die meist formale Beschreibung der Rechteprüfung und definiert zugleich die zu speichernden Entitäten für die Rechteverwaltung. Die Rechteprüfung wird vom Zugriffskontrollmechanismus realisiert und ist nach dem ISO Access Framework umzusetzen. Nach der Rechteprüfung wird jede Anfrage auf Zugriff protokolliert.

Aus den Entitäten Objekt, Operator, Zugriffsrecht, Subjekt, Sitzung und der auf Modellebene zu berücksichtigenden Konstruktionsprinzipien für sicherere Zugriffskon-

trollsysteme wurde ein Klassifikationsrahmen entwickelt. In diesen wurden die analysierten Zugriffskontrollmodelle Zugriffsmatrix, Bell-LaPadula-Modell, BIBA-Integritätsmodell, Clark-Wilson-Modell, Chinese-Wall-Modell, Verbandsmodell und rollenbasiertes Zugriffskontrollmodell eingeordnet und gegenübergestellt. Die Untersuchung ergab, dass sowohl die Zugriffsmatrix als auch das rollenbasierte Zugriffskontrollmodell für ein Zugriffskontrollsystem für Anwendungssysteme im IS als geeignet identifiziert wurde.

Das Referenzmodell des rollenbasierten Zugriffskontrollmodells wurde als Grundlage ausgewählt und ist aus folgenden Gründen besser geeignet als die Zugriffsmatrix:

- Die unterstützte Zugriffskontrollstrategie ist rollenbasiert, damit wird eine aufgabenorientierte Modellierung der Berechtigungen für die Zugriffskontrolle vorgegeben.
- Eine Rolle ist eine Zusammenfassung von Zugriffsrechten. Die Zugriffsrechte, die zur Erledigung der Aufgaben notwendig sind, stehen erst durch die Aktivierung einer Rolle einem Subjekt zur Verfügung.
- Die Modellierung der Aufgabentrennung wird bereits von RBAC mit dem Konzept der sich ausschließenden Rollen umgesetzt. Die Aufgabentrennung kann im Rahmen der Subjektzuordnung, aber auch dynamisch bei der Aktivierung der Rollen berücksichtigt werden.
- Das Modell selbst fordert, bereits jeden Zugriff zu protokollieren.
- Durch das Konzept der Rolle ist eine aufgabenbezogene Modellierung der Berechtigungsprofile möglich.

Diese Arbeit greift die nach der Veröffentlichung des RBAC-Referenzmodells geübte Kritik in den Bereichen Administration, Delegation und Dynamisierung auf. Diese Konzepte wurden eingehend untersucht und bilden modifiziert einen Teil der Erweiterung des Referenzmodells.

Gegenstand des zweiten Teils der Arbeit ist die Analyse des Begriffs Rolle, deren methodische Einordnung ins IS sowie die Erweiterung des Referenzmodells von RBAC. Der Begriff Rolle wird in verschiedenen wissenschaftlichen Disziplinen unterschiedlich interpretiert, wodurch ein Homonymkonflikt entsteht. Es wurden die unterschiedlichen, aber nicht immer disjunkten Beschreibungen der Rolle in folgende fünf Rollenkonzepte eingeordnet: verhaltensorientiert, organisationsorientiert, aufga-

benorientiert, kompetenzorientiert und berechtigungsorientiert. In allen Rollenkonzepten ist die Rolle vom Typ Aufgabenträger. Das Konzept der Virtualisierung kann analog virtueller Betriebsmittel auf die Rolle übertragen werden. Eine Rolle hat alle Eigenschaften des Aufgabenträgers, ist aber nicht auf die Kapazität eines Aufgabenträgers beschränkt. Die Rolle übernimmt dabei als virtueller Aufgabenträger die Repräsentation personeller Aufgabenträger im IS und damit auch im Zugriffskontrollmodell.

Aus den einzelnen Rollenkonzepten entstand ein ganzheitliches Rollenkonzept, das um Anwendungssysteme erweitert werden konnte. Dabei wurde ersichtlich, dass Zugriffsrechte sich auf die von außen aufrufbaren Funktionen eines AwS beziehen. Durch das Einbinden der Rolle in das betriebliche Informationssystem und ADK-Modell wird gezeigt, dass sich die Rolle als virtueller Aufgabenträger zwischen der Aufgaben- und Aufgabenträgerebene befindet. Rollen beziehen sich auf Aufgaben und sind ein geeignetes Instrument, um Aufgabenträger mit Zugriffsrechten zu verbinden, damit diese ihre Aufgaben innerhalb eines IS durchführen und gleichzeitig die Informationssicherheit gewährleistet wird. Deshalb ergänzt die Rollenzuordnung die Unternehmensarchitektur nach der SOM-Methodik an der Schnittstelle zwischen Geschäftsprozess- und Aufgabenträgerebene. Da sich Rollen aus dem Geschäftsprozess entwickeln lassen, ist der Top-Down Ansatz geeignet, ein valides Rollenmodell zu erstellen. Das Metamodell der Rollenzuordnung erweitert das Metamodell der Aufgabenträgerzuordnung um die Beziehungen Aufgabe und Aufgabenträger mit Rolle. Die Rollen bündeln dabei die Zugriffsrechte, die einem Aufgabenträger bei teilautomatisierten Aufgaben auf Aufgabenträgerebene an der Kommunikationsschnittstelle die notwendigen Zugriffe gewährt.

Durch Untersuchung des Konzeptes der Rolle im Konzept des betrieblichen Informationssystems wurde nochmals untermauert, dass die Rolle und damit auch das rollenbasierte Zugriffskontrollmodell prädestiniert sind eine Zugriffskontrolle für Anwendungssysteme mit aufgabenbezogenen Rollen umzusetzen. Aus den Defiziten des Referenzmodells ergab sich, dass eine Erweiterung des rollenbasierten Zugriffskontrollmodells zu eRBAC notwendig wurde, um eine umfassende Zugriffskontrolle von teilautomatisierten Aufgaben in Anwendungssystemen sicherzustellen.

Das Referenzmodell RBAC wurde zu eRBAC weiterentwickelt. Das Referenzmodell wird dabei erweitert um Objekttypen, Rollentypen, Personalisierung sowie Administ-

ration, Delegation und Domänenbeschränkung. Die Elemente von eRBAC und die Zusammenhänge wurden grafisch dargestellt und formal beschrieben. Nachstehende bereits existierende Konzepte wurden dabei modifiziert übernommen:

- Zur besseren Übersichtlichkeit werden Zugriffsrechte, die von mehreren Rollen benötigt werden, in virtuellen Rollen gekapselt. Diese virtuellen Rollen werden nicht direkt Subjekten zugeordnet, sondern nur in die Rollenhierarchie eingebunden.
- Die Rollenhierarchie wird als Zugriffsvererbungshierarchie, den intensionalen Aspekt der Rollenhierarchie, interpretiert. Um die Aufgabentrennung sicherzustellen und zu verhindern, dass zwei sich ausschließende Rollen aktiviert werden, wird zusätzlich eine Aktivierungshierarchie modelliert.
- Für die Autorisierung der Rechteverwaltung werden Administrationsrollen gebildet, die von den Anwendungsrollen disjunkt sind. Um die Administration dezentral durchführen zu können, werden die folgenden drei Administrationsbereiche gebildet: Zugriffsrechts-, Rollen- und Subjektverwaltung. Die Zuständigkeiten innerhalb der Administrationsbereiche werden über Vorbedingungen organisiert. Diese Vorbedingungen beziehen sich auf Zugriffsrechte, Rollen und Subjekte, die jedoch in Pools unabhängig von den modellierten Zugriffsrechten, Rollen und der Rollenhierarchie zusammengefasst sind.
- Durch Delegation werden Zugriffsrechte auf Zeit an einen Aufgabenträger, ohne bei der Weitergabe direkt eine Administration einzuschalten, weitergegeben. Um die Sicherheitsstrategie des Unternehmens nicht zu verletzen, legt die Administration die delegierbaren Zugriffsreche und Rollen fest. Außerdem muss jede Delegation auch wieder zurückgenommen werden können. Deshalb ist es neben einer automatischen Rücknahme zu einem festgelegten Zeitpunkt auch möglich, die Delegation durch den Delegierenden selbst oder die Administration zurückzunehmen.
- Durch die Erweiterung um die Domänenbeschränkung ist es möglich, dass Subjekte mit denselben Rollen nur auf Informationen ihrer erlaubten Domänen zugreifen können.

Folgende Erweiterungen wurden für eRBAC neu eingeführt:

- Die Objekte wurden in Objekttypen eingeteilt, um zwischen dem Aufruf eines AwS und den von außen aufrufbaren Funktionen innerhalb des AwS zu unterscheiden. Durch die beiden Objekttypen Anwendung und Klasse konnte die Implementierung einer flexiblen Aufrufstruktur bei der Authentifizierung und Zugriffskontrolle des anwendungsübergreifenden Teils realisiert werden.
- Die Rollen wurden in Rollentypen eingeteilt, um getrennte Sicherheitsrichtlinien für die einzelnen Rollentypen hinterlegen zu können und damit die Administration zu erleichtern.
- Die Rolle wurde um das Konzept der Personalisierung erweitert. Eine Personalisierung ist erforderlich, wenn nach der Authentifizierung Informationen, die sich auf Entitäten des AwS beziehen, benötigt werden. Dafür wird für die Rolle hinterlegt, welches Datenobjekt aus dem Zielanwendungssystem für die Personalisierung herangezogen wird. Die Administration ordnet nach der Subjektzuordnung dem Aufgabenträger die Schlüsselattributwerte aus dem AwS zu. Beim Aufruf des AwS werden diese Attributwerte übergeben und die Informationen für die Weiterverarbeitung entsprechend ermittelt und ggf. eingeschränkt.

Der erste Teil des Proof of Concept ist die exemplarische Realisierung eines Zugriffskontrollmechanismus. Die Realisierung erfolgt mit dem Zugriffskontrollsystem FN2RBAC auf Grundlage von eRBAC. Dabei werden die Konzeption, Architektur und Funktionsweise der Authentifizierung (FN2AUTH) und Zugriffskontrolle (FN2RBAC) für das Prüfungsverwaltungssystem FlexNow in Kapitel 7 vorgestellt. Die Rechteprüfung wird dabei unabhängig und getrennt von der Implementierung der Geschäftslogik an zentraler Stelle im Zugriffskontrollmechanismus FN2RBAC-RP realisiert, wie es der Standard ISO Access Control Framework als Norm vorschreibt. Das Datenschema und die Beschreibung der implementierten Authentifizierungstypen finden sich im Anhang (siehe Kapitel A).

Als zweiter Teil des Proof of Concept wurde in einer Fallstudie ein Rollenmodell des Prüfungsprozesses aus der Aufgabenebene heraus entwickelt. Die ermittelten Zugriffsrechte wurden mit Rollen assoziiert. Zugriffsrechte für dieselben Aufgaben verschiedener Rollen wurden in virtuellen Rollen zusammengefasst. Aus den Anwendungsrollen und den virtuellen Rollen wurde exemplarisch eine Rollenhierarchie entwickelt.

Ausblick und zusätzlicher Forschungsbedarf

Durch die fortwährende Weiterentwicklung der IT-Technologien wird das Thema Informationssicherheit, insbesondere die Umsetzung eines einheitlichen Identity-Managements und einer einheitlichen Zugriffskontrolle die Forschung auch zukünftig beschäftigen. Zusätzlich ist die Komplexität der IT-Landschaft in den letzten Jahren deutlich gewachsen, z. B. werden immer mehr Anwendungssysteme innerhalb einer Organisation durch Webservices verbunden. Es befinden sich aber immer noch Anwendungssysteme im Einsatz, die eine proprietäre Authentifizierung und/oder Zugriffskontrolle durchführen. Dies führt zu einer Redundanz bei der Datenhaltung und die Administration der verschiedenen Authentifizierungs- und Zugriffskontrollsysteme führt zu einem erhöhten Aufwand. Um dies zu verhindern, werden Versuche unternommen, ein einheitliches Identity- und Rollenmanagement zu entwickeln. Das Zugriffskontrollsystem FN2RBAC kann Zugriffsrechte, Rollen und Subjekte für unterschiedliche Anwendungssysteme verwalten und stellt dafür bereits einen Webservice zur Verfügung. Damit können Anwendungssysteme ihre bisher proprietäre Zugriffskontrolle ändern und über FN2RBAC anfragen, ob ein Subjekt die notwendige Autorisierung besitzt.

Es existieren auch Forschungsarbeiten, die RBAC um Organisationsrollen erweitern. Organisationsrollen kapseln die Rollen der unterschiedlichen Anwendungssysteme innerhalb einer Organisation (Roeckle et al. 2000; Park et al. 2004). Inwieweit die Administration von FN2RBAC durch Organisationsrollen vereinfacht und übersichtlicher gestaltet werden kann, indem eRBAC um den Rollentyp Organisationsrollen und einem Rollenmapping erweitert wird, bedarf noch weiterer Untersuchungen.

Die konkreten Rollen in der Fallstudie (siehe Kapitel 8) wurden in einem Role Engineering Prozess ermittelt, wohl wissend, dass auch Kritik am Top-Down Ansatz existiert und es Forschungsarbeiten gibt, die den Role-Mining Ansatz favorisieren (Kuhlmann et al. 2003; Kern et al. 2002; Vaidya et al. 2006). Role Mining versucht aus bestehenden Identity- und Zugriffskontrollsystemen die zu modellierenden Rollen zu extrahieren. Andere versuchen, um ein Rollenmodell der Organisation zu erhalten, diese beiden Ansätze zu einem hybriden Konzept zu verbinden (Fuchs und Pernul 2008) oder Rollen aus den mit BPEL4WS[72] definierten Prozessen zu extrahieren (Mendling et al. 2004). Hier müsste überprüft werden, ob der Ansatz, aus der

[72] BPEL4WS ist die Abkürzung von Business Process Execution Language for Web Services

Aufgabenebene das Rollenmodell zu entwickeln durch einen Role Mining Ansatz bestätigt oder unterstützt werden kann.

Der untersuchte Kontext des IS wurde eingeschränkt auf die Betrachtung innerhalb eines Unternehmens. Durch zunehmend gemeinschaftlich geprägte Arbeitsabläufe werden Informationen aus dem AwS auch organisationsübergreifend benötigt. Um diese Informationen austauschen zu können ist u. a. eine verteilte dezentrale Authentifizierung und Attributverwaltung auch über Organisationsgrenzen hinweg erforderlich. Eine Implementierungsmöglichkeit dafür ist Shibboleth® (Shibboleth 2015). Durch Übermittlung von Zertifikaten können Authentifizierungs- und Autorisierungsdaten auch anderen Organisationen zur Verfügung gestellt werden. Diese Möglichkeit wird unter dem Schlagwort attributbasierte Zugriffskontrolle (ABAC) untersucht (Li et al. 2002). Die Rolle kann dabei als eines der mitgelieferten Attribute definiert werden. Auch eine Parametrisierung und automatische Subjektzuordnung kann anhand von Attributen des Aufgabenträgers umgesetzt werden. Durch Übermittlung der notwendigen Daten in Zertifikaten könnte erreicht werden, dass auf eine Verwaltung der Subjekte für jedes Anwendungssystem verzichtet werden kann. Da die Rolle als virtueller Aufgabenträger betrachtet wird, kann an Hand des Attributs Rolle eine Zugriffskontrolle und zusammen mit den Attributen des Nutzers eine Personalisierung und Domänenbeschränkung durchgeführt werden. Das in dieser Arbeit entwickelte eRBAC Modell stellt alle dafür erforderlichen Entitäten und Konzepte bereits zur Verfügung. Lediglich die Implementierung müsste dafür entsprechend angepasst werden.

Literaturverzeichnis

Abadi M, Burrows M, Lampson B, Plotkin G (1993) A calculus for access control in distributed systems. ACM Transaction on Programming Languages and Systems 15(4):706–734. http://doi.acm.org/10.1145/155183.155225.

Adams JK (2006) A Service-Centric Approach to a Parameterized RBAC Service. http://arxiv.org/ftp/cs/papers/0603/0603030.pdf. Abruf am 2014-08-17.

Ahn G, Sandhu RS (2000) Role-based authorization constraints specification. ACM Transactions on Information and System Security 3(4):207–226. http://doi.acm.org/10.1145/382912.382913.

Alisch K (2004) Gabler-Wirtschafts-Lexikon. Gabler, Wiesbaden.

Al-Kahtani MA (2003) A family of models for rule-based user-role assignment. http://profsandhu.com/dissert/diss-mohammad.pdf. Abruf am 2014-08-17.

Al-Kahtani MA, Sandhu RS (2002) A Model for Attribute-Based User-Role Assignment. In: 18th Annual Computer Security Applications Conference, Washington, DC, USA.

Al-Kahtani MA, Sandhu RS (2004) Rule-Based RBAC with Negative Authorization. In: 20th Annual Computer Security Applications Conference, Washington, DC, USA.

Amoroso EG (1994) Fundamentals of computer security technology. PTR Prentice Hall, Englewood Cliffs.

Anderson JP (1972) Computer Security Technology Planning Study. Volume II. http://seclab.cs.ucdavis.edu/projects/history/papers/ande72.pdf. Abruf am 2014-08-17.

Anderson R (2001) Security engineering. A guide to building dependable distributed systems, New York.

ANSI INCITS 359-2004 (2004) Role Based Access Control. American National Standard for Information Technology. http://www.cs.purdue.edu/homes/ninghui/readings/AccessControl/ANSI+INCITS+359-2004.pdf. Abruf am 2014-08-17.

ATIS Telecom Glossary (2007) Separation of Duty. http://www.atis.org/glossary/definition.aspx?id=1872. Abruf am 2014-08-17.

Bakdi I (2007) Benutzerauthentifizierung anhand des Tippverhaltens bei Verwendung fester Eingabetexte. Univ., Diss.--Regensburg, 2007. Univ.-Verl., Regensburg.

Barka E, Sandhu RS (2000a) A Role-Based Delegation Model and Some Extensions. http://profsandhu.com/confrnc/nissc/rbdm00.pdf. Abruf am 2014-08-17.

Barka E, Sandhu RS (2000b) Framework for role-based delegation models. In: 16th Annual Computer Security Applications Conference, IEEE Computer Society, Washington, DC, USA.

Barka E, Sandhu RS (2004) Role-Based Delegation Model/ Hierarchical Roles (RBDM1). In: 20th Annual Computer Security Applications Conference, Washington, DC, USA.

Barkley JF (1997) Comparing simple role based access control models and access control lists. In: Second ACM workshop on Role-based access control, New York, NY, USA.

Bauknecht K, Holbein R (1996) Workflow-Management-Systems: Source and Solution of Privacy Problems in Organisations. In: Österle H (Hrsg.) Praxis des Workflow-Managements. Vieweg, Braunschweig.

Becker M (1998) Umsetzung betrieblicher Prozesse. Difo-Druck, Bamberg.

Bell ED (2005) Looking back at the Bell-La Padula Model. http://www.acsac.org/2005/papers/Bell.pdf. Abruf am 2014-08-17.

Bell ED, LaPadula LJ (1973a) Secure Computer Systems: A Mathematical Foundations. An electronic reconstruction by Len LaPadula of the original MITRE Technical Report 2547, Volume I titled “Secure Computer Systems: Mathematical Foundations” by D. Elliott Bell and Leonard J. LaPadula dated 1 March 1973. http://www.albany.edu/acc/courses/ia/classics/belllapadula1.pdf. Abruf am 2014-08-17.

Bell ED, LaPadula LJ (1973b) Secure Computer Systems: A Mathematical Model. An electronic reconstruction by Len LaPadula of the original MITRE Technical Report 2547, Volume II titled “Secure Computer Systems: Mathematical Foundations” by Leonard J. LaPadula and D. Elliott Bell dated 31 May 1973. http://www.albany.edu/acc/courses/ia/classics/belllapadula2.pdf. Abruf am 2014-08-17.

Bell ED, LaPadula LJ (1976) Secure Computer System: Unified Exposition and Multics Interpretation. http://csrc.nist.gov/publications/history/bell76.pdf. Abruf am 2014-08-17.

Beresnevichiene Y (2003) A role and context based security model. Technical Report Number 558. http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-558.pdf. Abruf am 2014-08-17.

Bertino E, Bonatti PA, Ferrari E (2001) TRBAC: A temporal role-based access control model. ACM Transactions on Information and System Security 4(3):191–233. http://doi.acm.org/10.1145/501978.501979.

Bertino E, Ferrari E, Atluri V (1997a) A flexible model supporting the specification and enforcement of role-based authorization in workflow management systems. In: Second ACM workshop on Role-based access control, New York, NY, USA.

Bertino E, Samarati P, Jajodia S (1993) Authorizations in Relational Database Management Systems. In: First ACM Conference on Computer and Communications Security, Fairfax, Virginia, United States.

Bertino E, Samarati P, Jajodia S (1997b) An Extended Authorization Model for Relational Databases. IEEE Transactions on Knowledge and Data Engineering 9(1):85–101. http://dx.doi.org/10.1109/69.567051.

bfdi (2010) Bundesdatenschutzgesetz (BDSG). http://www.bfdi.bund.de/DE/GesetzeUndRechtsprechung/BDSG/BDSG_node.html. Abruf am 2014-08-17.

Biddle BJ, Thomas EJ (Hrsg) (1966) Role theory. Concepts and research. John Wiley & Sons, New York.

Bill R, Zehner ML (2001) Zugriffskontrolle - Geoinformatik Lexikon. http://www.geoinformatik.uni-rostock.de/einzel.asp?ID=1806. Abruf am 2014-08-17.

Biltzinger P, Bunz H (2004) Erarbeitung einer Strategie zur Einführung der Gesundheitskarte. http://www.inso.tuwien.ac.at/uploads/media/b4h_sicherheitsarchitektur_v1-1.pdf. Abruf am 2014-08-17.

Bishop M (1981) Hierarchical Take-Grant Protection systems. SIGOPS Operating Systems Review 15(5):109–122. http://doi.acm.org/10.1145/1067627.806598.

Bishop M, Snyder L (1979) The transfer of information and authority in a protection system. In: 7th ACM symposium on Operating systems principles, Pacific Grove, California, United States.

Bleicher K (1980a) Kompetenz. In: Grochla E (Hrsg.) Handwörterbuch der Organisation. Poeschel, Stuttgart.

Bleicher K (1980b) Verantwortung. In: Grochla E (Hrsg.) Handwörterbuch der Organisation. Poeschel, Stuttgart.

Bless R, Mink S, Conrad M, Kutzner K, Blaß E, Hof H, Schöße M (2005) Sichere Netzwerkkommunikation. Springer, Berlin [u.a.].

Blohm H (1977) Organisation, Information und Überwachung. Gabler, Wiesbaden.

Blümle E (1975) Stellvertretung. In: Gaugler E (Hrsg.) Handwörterbuch des Personalwesens. Poeschel, Stuttgart.

Boella G, van der Torre L, Verhagen H (2007) Roles, an interdisciplinary perspective. Applied Ontology 2(2):81–88.

Bokranz R, Kasten L (2001) Organisations-Management in Dienstleistung und Verwaltung. Gabler, Wiesbaden.

Box D, Ehnebuske D, Kakivaya G, Laymann A, Medelsohn N, Nielsen HF, Thattle S, Winer D (2000) Simple Object Access Protocol (SOAP) 1.1. http://www.w3.org/TR/2000/NOTE-SOAP-20000508/. Abruf am 2014-08-18.

Brewer DF, Nash MJ (1989) The Chinese Wall Security Policy. In: 1989 IEEE Symposium on Security and Privacy, Washington, DC, USA.

Bronner R (1992) Verantwortung. In: Frese E (Hrsg.) Handwörterbuch der Organisation. Poeschel, Stuttgart.

Buchmann J (2004) Einführung in die Kryptographie. Springer, Berlin [u.a.].

Bundesamt für Sicherheit in der Informationstechnik (BSI) (2011) IT-Grundschutz-Kataloge. Bundesanzeiger, Köln.

Bussler C (1998) Organisationsverwaltung in Workflow-Management-Systemen. Deutscher Universitäts Verlag, Wiesbaden.

Cao X, Iverson L (2006) Intentional Access Management: Making Access Control Usable for End-Users.

http://cups.cs.cmu.edu/soups/2006/proceedings/p20_cao.pdf. Abruf am 2014-08-18.

Castano S, Fugini M, Martella G, Samarati P (1995) Database security. Addison-Wesley, Wokingham.

Chandramouli R (2000) Business Process Driven Framework for defining an Access Control Service based on Roles and Rules. http://csrc.nist.gov/nissc/2000/proceedings/papers/047.pdf. Abruf am 2014-08-18.

Clark DD, Wilson DR (1987) A Comparison of Commercial and Military Computer Security Policies. In: 1987 IEEE Symposium on Security and Privacy, IEEE Computer Society.

Coyne EJ, Davis JM (2008) Role engineering for enterprise security management. Artech House, Boston.

Coyne EJ, Weil T (2008) An RBAC Implementation and Interoperability Standard: The INCITS Cyber Security 1.1 Model. IEEE Security and Privacy 6(1):84–87. http://dx.doi.org/10.1109/MSP.2008.2.

Crook R, Ince D, Nuseibeh B (2002) Towards an Analytical Role Modelling Framework for Security Requirements. In: 8 th International Workshop on Requirements Engineering: Foundation for Software Quality, Essen, Germany.

Denning DE (1976) A lattice model of secure information flow. Communication of the ACM 19(5):236–243. http://doi.acm.org/10.1145/360051.360056.

Diffie W, Hellman M (1976) New directions in cryptography. IEEE Transactions on Information Theory 22(6):644–654.

Eckert C (2001) IT-Sicherheit. Oldenbourg, München.

Eckert C (2012) IT-Sicherheit. Oldenbourg, München.

Ehmann E (1993) Rechtliche Aspekte. In: Pohl H, Weck G (Hrsg.) Einführung in die Informationssicherheit. Oldenbourg, München, Wien.

Eren E, Detken K (2006) Mobile Security. Risiken mobiler Kommunikation und Lösungen zur mobilen Sicherheit. Hanser, München.

Essmayr W, Probst S, Weippl E (2004) Role-Based Access Controls: Status, Dissemination, and Prospects for Generic Security Mechanisms. Electronic Commerce Research 4(1-2):127–156.

Esswein W (1992) Das Rollenmodell der Organisation. Die Berücksichtigung aufbauorganisatorischer Regelungen in Unternehmensmodellen. Otto-Friedrich-Universität, Bamberg.

Eymann T (2013) Cloud Computing — Enzyklopaedie der Wirtschaftsinformatik. http://www.enzyklopaedie-der-wirtschaftsinformatik.de/wi-enzyklopaedie/lexikon/uebergreifendes/Kontext-und-Grundlagen/Markt/Softwaremarkt/Geschaftsmodell-%28fur-Software-und-Services%29/Cloud-Computing/index.html. Abruf am 2014-01-13.

Feng X, Guoyuan L, Hao Huang, Li Xie (2004) Role-Based Access Control System for Web Services. In: Fourth International Conference on Computer and Information Technology, Washington, DC, USA.

Ferraiolo DF, Barkley JF, Kuhn DR (1999) A role-based access control model and reference implementation within a corporate intranet. ACM Transactions on Information and System Security 2(1):34-64.

Ferraiolo DF, Cugini JA, Kuhn DR (1995) Role Based Access Control (RBAC): Features and Motivations. In: 11th Annual Computer Security Applications Conference, IEEE Computer Society.

Ferraiolo DF, Kuhn DR (1992) Role-based access control. In: 15th National Computer Security Conference, NIST-NCSC.

Ferraiolo DF, Kuhn DR, Chandramouli R (2003) Role-based access control. Artech House, Boston.

Ferraiolo DF, Kuhn R, Sandhu RS (2007) RBAC Standard Rationale: Comments on "A Critique of the ANSI Standard on Role-Based Access Control". IEEE Security and Privacy 5(6):51–53.

Ferraiolo DF, Sandhu RS, Gavrila S, Kuhn DR, Chandramouli R (2001) Proposed NIST standard for role-based access control. ACM Transactions on Information and System Security 4(3):224-274.

Ferstl OK, Sinz EJ (1995) Der Ansatz des Semantischen Objektmodells (SOM) zur Modellierung von Geschäftsprozessen. Wirtschaftsinformatik 37(3):209–220.

Ferstl OK, Sinz EJ (2013) Grundlagen der Wirtschaftsinformatik. Oldenbourg, München.

Fielding RT (2002) Architectural Styles and the Design of Network-based Software Architectures. http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm. Abruf am 2014-08-18.

Finkenzeller K (2006) RFID-Handbuch. Hanser, München.

Fischer L (1992) Rollentheorie. In: Frese E (Hrsg.) Handwörterbuch der Organisation. Poeschel, Stuttgart.

Franke W, Dangelmaier W (2006) RFID - Leitfaden für die Logistik. Gabler, Wiesbaden.

Frese E (1992) Organisationstheorie. Gabler, Wiesbaden.

Frings S, Weisbecker A (1998) Für jeden die passende Rolle. it Management(7):18–25.

Fuchs L, Pernul G (2008) HyDRo - Hybrid Development of Roles. In: R. Sekar, Arun K. Pujari (Hrsg.) Information systems security: 4th international conference, Braunschweig.

Gaitanides M (1983) Prozeßorganisation. Vahlen, München.

Gallaher MP, O'Connor AC, Kropp B (2002) The Economic Impact of Role-Based Access Control. www.rti.org/pubs/Role-Based_Access.pdf. Abruf am 2014-08-28.

Galler J (1995) Metamodelle des Workflow-Managements. In: Scheer A (Hrsg.) Veröffentlichungen des Instituts für Wirtschaftsinformatik.

Galler J (1997) Vom Geschäftsprozeßmodell zum Workflow-Modell. Gabler, Wiesbaden.

Gasser M (1988) Building a secure computer system. Van Nostrand Reinhold, New York.

Ge M, Osborn SL (2004) A Design for Parameterized Roles. In: Csilla Farkas, Pierangela Samarati (Hrsg.) 18th Annual Conference on Data and Applications, Sitges, Catalonia, Spain.

Gebel G (2003) Roles and Access Management: Seeking a Balance between Roles and Rules. https://www.gartner.com/doc/1404872/roles-access-management-seeking-balance. Abruf am 2014-08-28.

Giuri L, Iglio P (1996) A Formal Model for Role-Based Access Control with Constraints. In: 9th IEEE workshop on Computer Security Foundations, Washington, DC, USA.

Giuri L, Iglio P (1997) Role templates for content-based access control. In: Second ACM workshop on Role-based access control, New York, NY, USA.

Gligor VD, Gavrila SI, Ferraiolo DF (1998) On the Formal Definition of Separation-of-Duty Policies and their Composition. In: 1998 Symposium on Security and Privacy, IEEE Computer Society.

Goguen JA, Meseguer J (1982) Security Policies and Security Models. In: 1982 IEEE Symposium on Security and Privacy, IEEE Computer Society.

Gola P, Jaspers A (2002) Das neue BDSG im Überblick. Information zum BDSG 2001 bei Anwendung in der Privatwirtschaft. Datakontext-Fachverlag, Frechen.

Graf A (2002) Performance Measurement und Competency Management in der Praxis. HMD - Praxis Wirtschaftsinform.(227):46–55.

Graham GS, Denning PJ (1971) Protection: principles and practice. In: AFIPS ‘71 fall joint computer conference, New York, NY, USA.

Hagström A, Jajodia S, Parisi-Presicce F, Wijesekera D (2001) Revocations-A Classification. In: 14th IEEE workshop on Computer Security Foundations, Washington, DC, USA.

Hahn D (1975) Kompetenz. In: Gaugler E (Hrsg.) Handwörterbuch des Personalwesens. Poeschel, Stuttgart.

Haller NM (1994) The S/KEY TM one-time password system. In: 1994 Internet Society Symposium on Network and Distributed Systems.

Haller NM, Metz C, Nesser P, Straw M (1998) A One-Time Password System. http://tools.ietf.org/html/rfc2289. Abruf am 2014-08-28.

Harrison MA, Ruzzo WL, Ullman JD (1976) Protection in operating systems. Communication of the ACM 19(8):461-471.

Heilmann H (1996) Die Integration der Aufbauorganisation in Workflow-Management-Systeme. In: Heilmann H, Heinrich Lutz, Roithmayr F (Hrsg.) Information Engineering. Oldenbourg, München.

Herwig V, Schlabitz L (2004) Unternehmensweites Berechtigungsmanagement. Wirtschaftsinformatik 46(4):289–294.

Hoffmann F (1976) Entwicklung der Organisationsforschung. Gabler, Wiesbaden.

Hollingsworth D (2004) The Workflow Reference Model: 10 Years On. In: Fujitsu Services, UK; Technical Committee Chair of WfMC.

Hopcroft JE, Motwani R, Ullman JD (2002) Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie. Pearson Studium, München.

Hühnlein D, Korte U (2006) Grundlagen der elektronischen Signatur. Recht - Technik - Anwendung. SecuMedia-Verl., Ingelheim.

International Organization for Standardization (ISO) (2005) Information technology - Security techniques - Code of practice for information security management. ISO copyright Office, Genf 17799:2005(ISO/IEC 17799:2005).

Jablonski S (1995) Workflow-Management-Systeme. Modellierung und Architektur. Internat. Thomson Publ., Bonn.

Jablonski S, Böhm M, Schulze W (1997) Workflow-Management. Entwicklung von Anwendungen und Systemen. dpunkt.verlag, Heidelberg.

Jaeger T, Tidswell JE (2000) Rebuttal to the NIST RBAC model proposal. In: 5th ACM workshop on Role-based access control, New York, NY, USA.

Jajodia S (1997) Extended Authorization Model for Relational Databases. IEEE Transactions on Knowledge and Data Engineering 9:85–101.

Jajodia S, Samarati P, Sapino ML, Subrahmanian VS (2001) Flexible support for multiple access control policies. ACM Transactions on Database Systems 26(2):214-260.

Jajodia S, Samarati P, Subrahmanian VS (1997) A Logical Language for Expressing Authorizations. In: 1997 IEEE Symposium on Security and Privacy, Washington, DC, USA.

Janetzke P (2001) Flexibles und regelbasiertes Workflow-Management an Universitäten. Kovač, Hamburg.

Joshi JBD, Bertino E, Latif U, Ghafoor A (2005) A Generalized Temporal Role-Based Access Control Model. IEEE Transactions on Knowledge and Data Engineering 17(1):4–23.

Junk K, Mayer M (2003) Active Datamanagement. Säulen der Informationssicherheit. VDE-Verl., Berlin.

Karbe B (1994) Flexible Vorgangssteuerung mit ProMINanID. In: Hasenkamp U, Kirn S, Syring Michael (Hrsg.) CSCW - Computer supported cooperative work. Addison-Wesley, Bonn.

Kerckhoffs A (1883) La cryptographie militaire. Journal des sciences militaires 9(Jan):5–38.

Kern A (2002) Advanced Features for Enterprise-Wide Role-Based Access Control. In: 18th Annual Computer Security Applications Conference, Washington, DC, USA.

Kern A, Kuhlmann M, Kuropka R, Ruthert A (2004) A meta model for authorisations in application security systems and their integration into RBAC administration. In: 9th ACM symposium on Access control models and technologies, New York, NY, USA.

Kern A, Kuhlmann M, Schaad A, Moffett J (2002) Observations on the Role Life-cycle in the Context of Enterprise Security Management. In: 7th ACM Symposium on Access Control Models and Technologies, New York, NY, USA.

Kern A, Schaad A, Moffett JD (2003) An administration concept for the enterprise role-based access control model. In: 8th ACM symposium on Access Control Models and Technologies, New York, NY, USA.

Kern A, Walhorn C (2005) Rule support for role-based access control. In: 10th ACM symposium on Access Control Models and Technologies, New York, NY, USA.

Kersten H (1993) Internationale Sicherheitskriterien. Oldenbourg, München.

Kessler V (1992) Über Sinn und Unsinn von Sicherheitsmodellen. Datenschutz und Datensicherheit 16(9):462–466.

Kieser A, Walgenbach P (2003) Organisation. Schäffer-Poeschel, Stuttgart.

Kirn S, Kümmering U (1997) Organisatorische Perspektiven beim Einsatz von Workflow-Management Systemen. In: Ortner E (Hrsg.) Workflow-Management-Systeme im Spannungsfeld einer Organisation Proceedings EMISA-Fachgruppentreffen, Darmstadt.

Klippert H (2000) Methoden-Training. Übungsbausteine für den Unterricht. Beltz, Weinheim.

Kosiol E (1976) Organisation der Unternehmung. Gabler, Wiesbaden.

Krumbiegel J (1997) Integrale Gestaltung von Geschäftsprozessen und Anwendungssystemen in Dienstleistungsbetrieben. Deutscher Universitäts Verlag, Wiesbaden.

Kruth W (2001) IT-Grundlagenwissen für Datenschutz- und Sicherheitsbeauftragte in Wirtschaft und Verwaltung. Datakontext, Frechen.

Kuhlen R, Seeger T, Strauch D (Hrsg) (2004) Grundlagen der praktischen Information und Dokumentation. Saur., München.

Kuhlmann M, Shohat D, Schimpf G (2003) Role Mining - Revealing Business Roles for Security Administration Using Data Mining Technology. In: 8th ACM Symposium on Access Control Models and Technologies, New York, NY, USA.

Kuhn DR (1997) Mutual exclusion of roles as a means of implementing separation of duty in role-based access control systems. In: Second ACM workshop on Role-based access control, New York, NY, USA.

Kuhn DR (1998) Role based access control on MLS systems without kernel changes. In: Third ACM workshop on Role-based access control, New York, NY, USA.

Kuppinger M (2000) Microsoft Windows 2000 Server - das Handbuch. Microsoft Press, Unterschleißheim.

Kurz J (1998) Die Modellierung und Steuerung eines Kundenauftrags-Workflow im SAP R/3 mit Hilfe des SAP Business Workflow anhand eines Praxisbeispiels. Diplomarbeit Nr. 1632.

Lampson BW (1974) Protection. SIGOPS Operating Systems Review 8(1):18-24.

Landwehr CE (1981) Formal Models for Computer Security. ACM Computing Surveys 13(3):247-278.

Landwehr CE, Heitmeyer CL, McLean J (1984) A security model for military message systems. ACM Transactions on Computer Systems 2(3):198-222.

Lau B, Gerhardt W (1994) Ein rollenbasiertes unternehmensbezogenes Rechteverwaltungs-Paradigma. In: Bauknecht K, Teufel S (Hrsg.) Sicherheit in Informationssystemen, Zürich.

Lehmann FR (1999) Fachlicher Entwurf von Workflow-Management-Anwendungen. Teubner, Stuttgart.

Lehmann K (2007) Modelle und Techniken für eine effiziente und lückenlose Zugriffskontrolle in Java-basierten betrieblichen Anwendungen. Deutsche Nationalbibliothek, Frankfurt.

Li N, Byun J, Bertino E (2007) A Critique of the ANSI Standard on Role-Based Access Control. IEEE Security and Privacy 5(6):41-49.

Li N, Mitchell JC, Winsborough WH (2002) Design of a role-based trust-management framework. In: 2002 IEEE Symposium on Security and Privacy, Los Alamitos, Calif.

Liebrand M, Ellis HJC, Phillips CE, Demurjian SA, Ting TC (2002) Role Delegation for a Resource-Based Security Model. In: Ehud Gudes, Sujeet Shenoi (Hrsg.) 16th International Conference on Data and Applications Security, Kings College, Cambridge, UK.

Linkies M, Off F (2006) Sicherheit und Berechtigungen in SAP-Systemen. Galileo Press, Bonn.

Lipton RJ, Snyder L (1977) A Linear Time Algorithm for Deciding Subject Security. Journal of the ACM 24(3):455-464.

Manecke H (2004) Klassifikation, Klassieren. In: Kuhlen R, Seeger T, Strauch D (Hrsg.) Grundlagen der praktischen Information und Dokumentation. Saur., München.

Mayntz R (1980) Rollentheorie. In: Grochla E (Hrsg.) Handwörterbuch der Organisation. Poeschel, Stuttgart.

McLean J (1990) The Specification and Modeling of Computer Security. Computer 23(1):9-16.

Mendling J, Strembeck M, Stermsek G, Neumann G (2004) An Approach to Extract RBAC Models from BPEL4WS Processes. In: 13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, Washington, DC, USA.

Merkle RC (1978) Secure communications over insecure channels. Communication of the ACM 21(4):294-299.

Merton RD (1957) The Role-Set: Problems in Sociological Theory. British Journal of Sociology 8(2):106–120.

Metzger J, Siller H (2014) Gabler Wirtschaftslexikon, Stichwort: PIN-TAN-Verfahren. http://wirtschaftslexikon.gabler.de/Archiv/6361/pin-tan-verfahren-v6.html. Abruf am 2014-08-30.

Moffett JD (1998) Control principles and role hierarchies. In: Third ACM workshop on Role-based access control, New York, NY, USA.

Moffett JD, Lupu EC (1999) The uses of role hierarchies in access control. In: 4th ACM workshop on Role-based access control, New York, NY, USA.

Moschgath M (2002) Kontextabhängige Zugriffskontrolle für Anwendungen im Ubiquitous Computing. http://tuprints.ulb.tu-darmstadt.de/333/. Abruf am 2014-08-30.

Mühlen M von, Rosemann M (1996) Der Lösungsbeitrag von Metadatenmodellen beim Vergleich von Workflowmanagementsystemen. In: Becker J, Rosemann M (Hrsg.) Workflowmanagement – State-of-the-Art aus Sicht von Theoris und Praxis.

Müller BF, Stolp P (1999) Workflow-Management in der industriellen Praxis. Springer-Verlag, Berlin.

Müller K (2005) IT-Sicherheit mit System. Vieweg, Wiesbaden.

Murauer J (2001) Informationsflussorientierte Verfahren zum Zugriffsschutz in Computersystemen. Österreichische Disserdationsdatenbank.

Na S, Cheon S (2000) Role delegation in role-based access control. In: 5th ACM workshop on Role-based access control, New York, NY, USA.

Nash MJ, Poland KR (1990) Some Conundrums Concerning Separation of Duty. In: 1990 IEEE Symposium on Security and Privacy, IEEE Computer Society, Washington, DC, USA.

NCSC (1985) DoD Trusted Computer System Evaluation Criteria. http://csrc.nist.gov/publications/history/dod85.pdf. Abruf am 2014-08-30.

NCSC (1988) Glossary of Computer Security Terms. http://fas.org/irp/nsa/rainbow/tg004.htm. Abruf am 2014-08-30.

Neumann G, Strembeck M (2001) Design and implementation of a flexible RBAC-service in an object-oriented scripting language. In: 8th ACM conference on Computer and Communications Security, New York, NY, USA.

Neumann G, Strembeck M (2002) A Scenario-driven Role Engineering Process for Functional RBAC Roles. In: 7th ACM Symposium on Access Control Models and Technologies, New York, NY, USA.

Nordsieck F (1972) Betriebsorganisation. Betriebsaufbau u. Betriebsablauf. Poeschel, Stuttgart.

Nyanchama M, Osborn SL (1994) Access Rights Administration in Role-Based Security Systems. In: FIP WG11.3 Working Conference on Database Security VII, Amsterdam, The Netherlands.

Nyanchama M, Osborn SL (1996) Modeling mandatory access control in role-based security systems. In: 9th annual IFIP TC11 WG11.3 working conference on Database security IX, London, UK, UK.

Nyanchama M, Osborn SL (1999) The role graph model and conflict of interest. ACM Transactions on Information and System Security 2(1):3-33.

o. V. (2014a) Wie kann ich das Verhalten der Benutzerkontensteuerung mithilfe von Gruppenrichtlinien ändern? http://windows.microsoft.com/de-de/windows7/how-do-i-change-the-behavior-of-user-account-control-by-using-group-policy. Abruf am 2015-02-21.

o.V. (2014b) Hibernate. Everything Data. http://www.hibernate.org/. Abruf am 2015-02-26.

Oh S, Sandhu RS (2002) A model for role administration using organization structure. In: 7th ACM symposium on Access control models and technologies, New York, NY, USA.

Oracle Corporation Java 2 Platform, Enterprise Edition (J2EE) Overview. http://www.oracle.com/technetwork/java/javaee/overview/index.html. Abruf am 2014-08-30.

Osborn SL, Sandhu RS, Munawer Q (2000) Configuring role-based access control to enforce mandatory and discretionary access control policies. ACM Transactions on Information and System Security 3(2):85-106.

Park JS, Costello KP, Neven TM, Diosomito JA (2004) A composite rbac approach for large, complex organizations. In: 9th ACM symposium on Access control models and technologies, New York, NY, USA.

Peacock A, Ke X, Wilkerson M (2004) Typing Patterns: A Key to User Identification. IEEE Security and Privacy 2(5):40–47.

Picot A, Dietl H, Franck E (1997) Organisation. Eine ökonomische Perspektive. Schäffer-Poeschel, Stuttgart.

Picot A, Reichwald R, Wigand RT (2003) Die grenzenlose Unternehmung. Information, Organisation und Management. Gabler, Wiesbaden.

Pohl H (2004) Taxonomie und Modellbildung in der Informationssicherheit. Datenschutz und Datensicherheit 28(11):678–685.

Pohl H, Weck G (Hrsg) (1993) Einführung in die Informationssicherheit. Oldenbourg, München, Wien.

Prescher J, Schefer-Wenzl S, Baumgrass A, Strembeck M, Mendling J (2014) Towards a Comprehensive Complexity Assessment of RBAC Models. EMISA Forum 34(2):12–23. http://www.emisa.org/images/Forum/2014-2/2014_2_paper1.pdf. Abruf am 2014-08-30.

Raepple M (2001) Sicherheitskonzepte für das Internet. dpunkt.verlag, Heidelberg.

Raev A, Schambeck M, Wagner-Braun M (Hrsg) (2010) Kolloquium 2010. Beiträge Bamberger Nachwuchswissenschaftlerinnen. Univ. of Bamberg Press, Bamberg.

Rankl W (2006) Chipkarten-Anwendungen. Hanser, München.

Rankl W (2008) Handbuch der Chipkarten. Hanser, München.

Rankl W, Effing W (2002) Handbuch der Chipkarten. Hanser, München.

Reeg T (2012) Modellierung betrieblicher Informationssicherheit. Entwicklung einer geschäftsprozessgetriebenen Modellierungsmethodik unter Nutzung eines Referenzmodells. Univ., Diss, Bamberg.

Reichenbach M (2004) Informationssicherheits-Management. In: Ernst S (Hrsg.) Hacker, Cracker & Computerviren. Recht und Praxis der Informationssicherheit. Schmidt, Köln.

Reichert M, Dadam P (2000) Geschäftsprozessmodellierung und Workflow-Management - Konzepte, Systeme und deren Anwendung. Industrie Management 16(3):23-27. http://dbis.eprints.uni-ulm.de/239/. Abruf am 2014-08-30.

Reichwald R (Hrsg) (2000) Telekooperation. Verteilte Arbeits- und Organisationsformen. Springer, Berlin.

Riechmann T (1999) Sicherheit in verteilten, objektorientierten Systemen. http://www4.informatik.uni-erlangen.de/Diss/pdf/PHD-Riechmann-A4.pdf. Abruf am 2014-08-30.

Rieger S (2007) Einheitliche Authentifizierung in heterogenen IT-Strukturen für ein sicheres e-Science-Umfeld. Cuvillier, Göttingen.

Rockart JF, Earl MJ, Ross JW (1996) Eight imperatives for The new IT organization. Sloan Management Review 38(1):43–55.

Roeckle H, Schimpf G, Weidinger R (2000) Process-oriented approach for role-finding to implement role-based security administration in a large industrial organization. In: 5th ACM workshop on Role-based access control, New York, NY, USA.

Rompel J (1990) One-way functions are necessary and sufficient for secure signatures. In: 22th annual ACM symposium on Theory of computing, New York, NY, USA.

Rosemann M, Mühlen M von (1997) Modellierung der Aufbauorganisation in Workflow-Management-Systemen: Kritische Bestandsaufnahme und Gestaltungsvorschläge. In: Ortner E (Hrsg.) Workflow-Management-Systeme im Spannungsfeld einer Organisation Proceedings EMISA-Fachgruppentreffen, Darmstadt.

Rühli E (1993) Unternehmungsführung und Unternehmungspolitik. Haupt, Bern.

Rupietta W, Wernke G (1994) Umsetzung organisatorischer Regelungen in der Vorgangsbearbeitung mit WorkParty und ORM. In: Hasenkamp U, Kirn S, Syring Michael (Hrsg.) CSCW - Computer supported cooperative work. Informationssysteme für dezentralisierte Unternehmensstrukturen. Addison-Wesley, Bonn.

Sackmann S (2012) IT-Sicherheit — Enzyklopaedie der Wirtschaftsinformatik. http://www.enzyklopaedie-der-wirtschaftsinformatik.de/wi-enzyklopaedie/lexikon/technologien-methoden/Informatik--Grundlagen/IT-Sicherheit/. Abruf am 2014-08-30.

Saltzer JH, Schroeder MD (1975) The protection of information in computer systems. Proceedings of the IEEE 63(9):1278-1308. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1451869. Abruf am 2014-08-30.

Samarati P, De Capitani dei Vimercati (2001) Access Control: Policies, Models, and Mechanisms. In: Focardi R (Hrsg.) Foundations of security analysis and design. Tutorial lectures. Springer, Berlin.

Sandhu RS (1988) Transaction control expressions for separation of duties. In: 4th Computer Security Applications Conference.

Sandhu RS (1991) Separation of duties in computerized information systems. In: Jajodia S, Landwehr CE (Hrsg.) Database Security IV: Status and Prospects.

Sandhu RS (1992) The Typed Access Matrix Model. In: 1992 IEEE Symposium on Security and Privacy, Washington, DC, USA.

Sandhu RS (1996) Roles versus groups. In: 1st ACM Workshop on Role-based access control, New York, NY, USA.

Sandhu RS (1998) Role activation hierarchies. In: Third ACM workshop on Role-based access control, New York, NY, USA.

Sandhu RS, Bhamidipati V, Munawer Q (1999) The ARBAC97 model for role-based administration of roles. ACM Transactions on Information and System Security 2(1):105-135.

Sandhu RS, Coyne EJ, Feinstein HL, Youman CE (1996) Role-Based Access Control Models. Computer 29(2):38–47.

Sandhu RS, Munawer Q (1999) The ARBAC99 Model for Administration of Roles. In: 15th Annual Computer Security Applications Conference, IEEE Computer Society, Washington, DC, USA.

SAP (2007) Rolle (SAP-Bibliothek - Integration zum SAP Business Workflow). http://help.sap.com/saphelp_46c/helpdata/de/bb/bdc296575911d189240000e8323d3a/content.htm. Abruf am 2014-09-03.

Saunders G, Hitchens M, Varadharajan V (2001) Role-based access control and the access control matrix. SIGOPS Operating Systems Review 35(4):6-20.

Schäfer G (2003) Netzsicherheit. Algorithmische Grundlagen und Protokolle. dpunkt.verlag, Heidelberg.

Scheinerman ER (2000) Mathematics. A discrete introduction. Brooks/Cole, Pacific Grove, Calif.

Schier K (1999) Vertrauenswürdige Kommunikation im elektronischen Zahlungsverkehr. http://www.informatik.uni-hamburg.de/bib/medoc/B-222.pdf. Abruf am 2014-09-03.

Schreyögg G (2008) Organisation. Grundlagen moderner Organisationsgestaltung. Gabler, Wiesbaden.

Schwarz H (1980) Aufgabenträger. In: Grochla E (Hrsg.) Handwörterbuch der Organisation. Poeschel, Stuttgart.

Schwenk J (2005) Sicherheit und Kryptographie im Internet. Vieweg, Wiesbaden.

Seufert SE (2001) Die Zugriffskontrolle. Otto-Friedrich-Universität, Bamberg.

Shibboleth (2015) Shibboleth. https://shibboleth.net/. Abruf am 2015-02-08.

Simon R, Zurko ME (1997) Separation of Duty in Role-based Environments. In: 10th IEEE workshop on Computer Security Foundations, Washington, DC, USA.

Sinz EJ (1995) Serviceorientierung der Hochschulverwaltung und ihre Unterstützung durch workflow-orientierte Anwendungssysteme. Vortrag, European University Information Systems, Congress. Otto-Friedrich-Universität, Bamberg.

Sinz EJ (1998a) Prozeßgestaltung und Prozeßunterstützung im Prüfungswesen. Otto-Friedrich-Universität, Bamberg.

Sinz EJ (1998b) Universitätsprozesse. In: Küpper H, Sinz EJ (Hrsg.) Gestaltungskonzepte für Hochschulen. Schäffer-Pöschel, Stuttgart.

Sinz EJ, Krumbiegel J (1996) Universitätsprozeß "Studium und Lehre". https://sedaintra.seda.wiai.uni-bamberg.de/forschung/kumi/lehre/lehre.htm. Abruf am 2014-09-04.

Sinz EJ, Krumbiegel J (2006) Universitätsprozess "Prüfung". https://sedaintra.seda.wiai.uni-bamberg.de/forschung/kumi/lehre/pruefung/pruefung.htm. Abruf am 2014-09-04.

Sinz EJ, Wismans B (1998) Das "Elektronische Prüfungsamt". Otto-Friedrich-Universität, Bamberg.

Snyder L (1981) Formal Models of Capability-Based Protection Systems. IEEE Transactions on Computers 30(3):172-181.

Spies PP (1985) Datenschutz und Datensicherung im Wandel der Informationstechnologien. 1. GI-Fachtagung München, 30. u. 31. Okt. 1985. Springer-Verlag, Berlin.

Staehle WH (1999) Management. Vahlen, München.

Steinmann H, Schreyögg G (2000) Management. Grundlagen der Unternehmensführung. Gabler, Wiesbaden.

Stiemerling O (2002) Web-Services als Basis für evolvierbare Softwaresysteme. Wirtschaftsinformatik 44(5):435–445.

Stiemerling O, Won M, Wulf V (2000) Zugriffskontrolle in Groupware - Ein nutzerorientierter Ansatz. Wirtschaftsinformatik 42(4):318–328.

Strasmann J (1996) Kernkompetenzen. Schäffer-Pöschel, Stuttgart.

Strembeck M, Neumann G (2004) An integrated approach to engineer and enforce context constraints in RBAC environments. ACM Transactions on Information and System Security 7(3):392-427.

Summers RC (1984) An Overview of Computer Security. IBM Systems Journal 23(4):309–325.

Thomas E, Biddle BJ (1966) The Nature and History of Role Theory. In: Biddle BJ, Thomas EJ (Hrsg.) Role theory. Concepts and research. John Wiley & Sons, New York.

Thomas RK, Sandhu RS (1993) Towards a task-based paradigm for flexible and adaptable access control in distributed applications. In: 1992-1993 workshop on New security paradigms, New York, NY, USA.

Thomas RK, Sandhu RS (1994) Conceptual Foundations for a Model of Task-based Authorizations. In: VII. Workshop: Computer Security Foundations, Los Alamitos, Calif.

Tinnefeld M, Ehmann E, Gerling RW (2005) Einführung in das Datenschutzrecht. Datenschutz und Informationsfreiheit in europäischer Sicht. Oldenbourg, München.

Türker C, Saake G (2006) Objektrelationale Datenbanken. dpunkt, Heidelberg.

Vaidya J, Atluri V, Warner J (2006) RoleMiner: Mining Roles Using Subset Enumeration. In: 13th ACM Conference on Computer and Communications Security, New York, NY, USA.

van Aalst der W, Hee van KM (2002) Workflow management. Models, methods, and systems. MIT Press, Cambridge, Mass.

Voßbein R (2005) Der Datenschutzbeauftragte. In: Voßbein R (Hrsg.) Die Organisation der Arbeit des betrieblichen Datenschutzbeauftragten. Datakontext-Fachverlag, Frechen.

W3C (1999) Forms in HTML documents. http://www.w3.org/TR/1999/REC-html401-19991224/interact/forms.html. Abruf am 2014-09-03.

W3C (2009) World Wide Web Consortium (W3C). http://www.w3.org/. Abruf am 2014-09-03.

Wahl M, Alvestrand H, Hodges J, Morgan R. (2000) Authentication Methods for LDAP. http://www.rfc-editor.org/rfc/rfc2829.txt. Abruf am 2014-09-03.

Wahl M, Howes T, Kille S (1997) Lightweight Directory Access Protocol. http://www.rfc-editor.org/rfc/rfc2251.txt. Abruf am 2014-09-03.

Walther I (2005) Rollen- und Situationsmodellierung bei betrieblichen Dispositions- und Planungssystemen. http://www.opus.ub.uni-erlangen.de/opus/volltexte/2005/137/. Abruf am 2014-09-03.

Weck G (1984) Datensicherheit. Methoden, Maßnahmen und Auswirkungen des Schutzes von Informationen. Teubner, Stuttgart.

Weck G (1993) Realisierung der Schutzfunktionen. In: Pohl H, Weck G (Hrsg.) Einführung in die Informationssicherheit. Oldenbourg, München, Wien.

Wermke M, Kunkel-Razum K, Scholze-Stubenrecht W (2006) Duden - die deutsche Rechtschreibung. Dudenverlag, Mannheim.

Weske M (1999) Workflow management systems. Formal foundation, conceptual design, implementation aspects. Univ. Habil.-Schr., Münster.

WfMC (1999) Terminology & Glossary. Document Number WfMC TC-1011. http://www.wfmc.org/standards/docs/TC-1011_term_glossary_v3.pdf. Abruf am 2015-02-21.

WfMC (2005) Interface 1 - Process Definition Interchange. http://www.wfmc.org/docs/TC-1025_xpdl_2_2005-10-03.pdf. Abruf am 2015-02-21.

Zhang L, Ahn G, Chu B (2001) A rule-based framework for role based delegation. In: 6th ACM symposium on Access control models and technologies, New York, NY, USA.

Zhang L, Ahn G, Chu B (2003a) A rule-based framework for role-based delegation and revocation. ACM Transactions on Information and System Security 6(3):404-441.

Zhang X, Li Y, Nalla D (2005) An attribute-based access matrix model. In: 2005 ACM symposium on Applied computing, New York, NY, USA.

Zhang X, Oh S, Sandhu RS (2003b) PBDM: a flexible delegation model in RBAC. In: 8th ACM symposium on Access control models and technologies, New York, NY, USA.

zur Muehlen M (2004) Organizational Management in Workflow Applications - Issues and Perspectives. Information Technology and Management 5(3):271–291.

A Anhang: Implementierung von FN2AUTH und FN2RBAC

Nach der Beschreibung der Konzeption, Architektur und Funktionsweise von FN2AUTH und FN2RBAC in Kapitel 7 werden in diesem Kapitel die implementierten Authentifizierungstypen und die Datenschemata für die Authentifizierung, Zugriffskontrolle und Protokollierung vorgestellt.

A.1 Implementierung von FN2AUTH

Die Implementierung der Authentifizierung sollte unabhängig von den eigentlichen Anwendungen sein, damit diese für das gesamte Unternehmen eingesetzt werden kann. Eine Authentifizierung mit verschiedenen Authentifizierungsmerkmalen und Verfahren (Kapitel 2.3.2 und Kapitel 7.2) wird durch das Konzept der Authentifizierungstypen umgesetzt.

A.1.1 Authentifizierungstypen in FN2AUTH

Folgende Authentifizierungstypen wurden in FN2AUTH implementiert:

- Zugangsdaten lokal prüfen: Die eingegebenen Zugangsdaten werden in der Datenbank FN2META geprüft.
- LDAP: Zugangsdaten werden direkt in einem LDAP-Verzeichnis (Wahl et al. 1997; Wahl et al. 2000) geprüft.
- REMOTEUSER: Das Login wird durch einen Applikationsserver geprüft. Nach erfolgreicher Authentifizierung wird eine Umgebungsvariable des Applikationsservers der sog. **REMOTEUSER** gesetzt, der dann von einem Authentifizierungssystem abgefragt wird.
- ZERTIFIKAT: Ein Applikationsserver überprüft, ob an dem Rechner eine gültige Chipkarte angeschlossen ist. Er liest das auf der Karte gespeichert Zertifikat aus und stellt es für die Authentifizierung in der Umgebungsvariable **X509Certificate** des Applikationsservers zur Verfügung. Im Zertifikat ist bei erfolgreicher Authentifizierung eine eindeutige festgelegte Kennung hinterlegt. Mit dieser Kennung kann FN2AUTH das entsprechende Subjekt ermitteln.
- AUTHTYP_AUTHCODE: Von der existierenden Anwendung, die eine eigenständige Authentifizierung durchführt, wird ein Token erzeugt und

übergeben. Dieses Token beinhaltet eine verschlüsselte Information, anhand derer überprüft werden kann, ob ein Studierender bereits authentifiziert wurde. Durch den in der Datenbank hinterlegten Token kann FN2AUTH feststellen, um welchen Studierenden es sich handelt und die Personalisierung vornehmen.

Nachfolgend wird eine kurze Beschreibung der unterstützten Verfahren für ausgewählte Authentifizierungstypen gegeben:

LDAP

Das Lightweight Directory Access Protokoll (LDAP) ist in der Version 3 im Standard RFC-2251 beschrieben. LDAP stellt ein Abfrageprotokoll für X.500-Verzeichnisse dar. Der X.500 Standard wird mit anderen Technologien zu netzwerkbasierten Verzeichnissen im internationalen Standard im ISO9594 zusammengefasst. Das X.500 Verzeichnis organisiert große Datenmengen bei Bedarf verteilt über mehrere Rechner in einer hierarchischen Baumstruktur. Es bestehen zahlreiche Möglichkeiten, diese zu durchsuchen. Für LDAP stehen z. B. Programmierschnittstellen für Java, C und C++ sowie Implementierungen in Applikationsservern zur Verfügung[73].

Web- bzw. Applikationsserver

Viele Applikationsserver bieten alleine oder in Zusammenarbeit mit einem Webserver die Möglichkeit, eine Authentifizierung deklarativ durchzuführen. Für die Authentifizierungstypen REMOTEUSER und ZERTIFIKAT wird auf diese Möglichkeit zurückgegriffen. Durch Konfiguration des Applikationsservers können verschiedene Varianten eingestellt werden[74]. Folgende Möglichkeiten der Authentifizierung werden von Applikationsservern u. a. unterstützt:

- Authentifizierung über XML-Dateien
- Authentifizierung über Chipkarten mit Zertifikaten
- Authentifizierung über eine Datenbank
- Authentifizierung über LDAP

Der Vorteil, die Überprüfung der Identität in den Applikationsserver zu verlegen, liegt darin, dass standardisierte Verfahren eingesetzt werden können. Lösungen auf

[73] Für weitere Informationen wird auf die Literatur verwiesen Eren und Detken (2006, S. 120–125).
[74] Für eine ausführliche Erklärung der möglichen Konfigurationen wird auf die entsprechenden Dokumentationen verwiesen.

Basis eines festgelegten, bekannten Standards sind proprietären Lösungen vorzuziehen (Eren und Detken 2006, S. 510). Darüber hinaus kann für Webanwendungen damit die Möglichkeit der Single-Sign-On-Authentifizierung bereitgestellt werden. Durch die Möglichkeit der deklarativen Authentifizierung über einen Applikationsserver und der Weiterverarbeitung dieser Information in FN2AUTH, werden die Vorteile der deklarativen und programmatischen Umsetzung verknüpft.

Zugangsdaten in einer Datenbank prüfen

Das implementierte Passwortverfahren zur Überprüfung kann in einer beliebigen Programmiersprache z. B. Java geschrieben werden. Die vom Subjekt eingegebenen Zugangsdaten werden mit den gespeicherten Informationen in einer Datenbank verglichen und bei Gleichheit ist die Identität des Subjekts festgestellt.

Proprietäre Verfahren

Zusätzlich wurde für das Prüfungsverwaltungssystem FlexNow ein proprietäres Verfahrens entwickelt. Dieses Verfahren wird mit dem Authentifizierungstyp AUTHTYP_AUTHCODE eingestellt.

A.1.2 Datenschema der Authentifizierung

Das Datenschema[75] für die Authentifizierung umfasst vier Tabellen:

- In der Tabelle **Authtyp** werden die Authentifizierungstypen gespeichert.
- In der Tabelle **Crypttyp** wird hinterlegt, mit welchen Verschlüsselungsalgorithmen das Passwort verschlüsselt ist. Dies ist nur relevant, wenn die Zugangsdaten direkt in einer Datenbank geprüft werden.
- In der Tabelle **Person** sind sämtliche Subjekte gespeichert, die über FN2AUTH authentifiziert werden.
- In der Tabelle **Auth** werden zu jeder Person alle erlaubten Authentifizierungstypen gespeichert.

Wie **Abb. A-1** zeigt, sind **Crypttyp**, **Authtyp** und **Person** nicht existenzabhängige Datenobjekttypen. Eine Person sind Null bis beliebig viele Einträge des existenzab-

[75] Für die Darstellung dieses konzeptionellen Datenschemas sowie aller weiteren Datenschemata wird das strukturierte Entity-Relationship-Modell (SERM) verwendet Ferstl und Sinz (2013, S. 168–171).

hängigen Gegenstands-Beziehungsobjekttyp Auth zugeordnet. Ein Eintrag in Auth bezieht sich auf genau eine Person, einen Authtyp und einen Crypttyp.

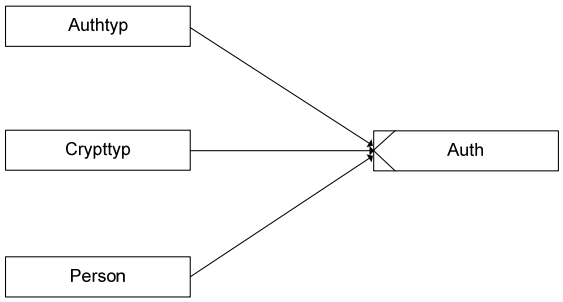

Abb. A-1 Datenschema der Authentifizierung

Sollte zur Authentifizierung ein externer Dienst wie LDAP oder Applikationsserver verwendet werden, sind dennoch Einträge in der Tabelle **Person** und **Auth** erforderlich. In diesem Fall werden keine Passwörter gespeichert, da diese vom externen Dienst vorgehalten werden. Anhand der Kennung in der Tabelle **Auth** wird die authentifizierte Person ermittelt, damit anschließend die Zugriffskontrolle durchgeführt kann.

A.2 Datenschema der Zugriffskontrolle – FN2RBAC

Die Dokumentation des Datenschemas für die Autorisierung wird gemeinsam für die Rechteverwaltung (FN2RBAC-V) und Rechteprüfung (FN2RBAC-RP) beschrieben und danach für die Protokollierung (FN2RBAC-P).

A.2.1 Datenschema von FN2RBAC

Das im folgendem beschriebene Datenschema der Zugriffskontrolle umfasst die Grundfunktionen Rechteverwaltung und Rechteprüfung und beschreibt damit ausschließlich den Bereich der Zugriffskontrolle.

Das Datenschema der Zugriffskontrolle umfasst die folgenden Tabellen:

- In der Tabelle **Person** sind sämtliche Subjekte gespeichert, die über FN2AUTH authentifiziert und über FN2RBAC autorisiert werden.
- In der Tabelle **Parameter** werden alle Tabellen aus dem Zielanwendungssystem hinterlegt, für die eine Domänenbeschränkung notwendig ist.
- In der Tabelle **Rollentyp** werden die verschiedenen Rollentypen hinterlegt.
- In der Tabelle **Objekttyp** werden die verschiedenen Objekttypen gespeichert.
- In der Tabelle **Operator** werden alle möglichen Operatoren hinterlegt.

- In der Tabelle **Rolle** werden alle Rollen gespeichert.
- In der Tabelle **Objekt** werden alle Objekte, die durch das Zugriffskontrollsystem geschützt werden müssen gespeichert.
- In der Tabelle **Personrolle** wird die Subjektzuordnung gespeichert. Sie verfügt über zusätzliche Attribute, die die Gültigkeitsdauer beschränken.
- In der Tabelle **Rolleparameter** werden einer Rolle die Parameter zugeordnet, für die eine Domänenbeschränkung vorgenommen werden muss.
- In der Tabelle **Rechte** werden die Zugriffsrechte gespeichert, indem einem Objekt die erlaubten Operatoren zugeordnet werden.
- Ist für eine Rolle eine Personalisierung vorgesehen, werden in der Tabelle **Keyattribut** die benötigten Schlüsselattribute hinterlegt.
- In der Tabelle **Personparameter** werden die Schlüsselattribute der Parameter zur Domänenbeschränkung hinterlegt.
- Eine statische bzw. dynamische Aufgabentrennung wird in der Tabelle **AT**[76] gespeichert. Hier werden sich ausschließende Rollen gespeichert und durch ein Flag gekennzeichnet, ob die Aufgabentrennung statisch oder dynamisch überwacht werden muss.
- In der Tabelle **Rollenrechte** wird die Zugriffsrechtszuordnung gespeichert.
- In der Tabelle **Rollenhierarchie** wird die Einordnung einer Rolle die Rollenhierarchie hinterlegt.

Die Abhängigkeiten der einzelnen Entitäten des Datenschema der Autorisierung (**Abb. A-2**) können wie folgt beschrieben werden: **Person, Parameter, Rollentyp, Objekttyp und Operator** sind originäre, nicht existenzabhängige Datenobjekttypen.

Ein **Objekttyp** hat null bis beliebig viele Objekte (**Objekt**) zugeordnet. Jedes **Objekt** bezieht sich auf genau einen **Objekttyp**. Einem **Objekt** sind null bis beliebig viele Operatoren (**Operator**) im Gegenstands-Beziehungsobjekttyp **Rechte** zugeordnet. Analog gilt dies ebenso für Operatoren. Zwischen **Objekt** und **Operator** besteht eine n:m Beziehung.

Einer **Rolle** können null bis beliebig viele sich ausschließende Rollen (**Rolle**) zugeordnet werden. Gespeichert wird dies im Beziehungsobjekttyp **AT**. Bezieht sich eine

[76] AT ist in diesem Kontext die Abkürzung von Aufgabentrennung.

Beziehung auf denselben Datenobjekttyp, so gehen von diesem Datenobjekttyp (**Rolle**) zwei Kanten zum gleichen Gegenstands-Beziehungsobjekttyp (**AT**). Eine **Rolle** kann null bis beliebig viele andere Rollen (**Rolle**) im Beziehungsobjekttyp **Rollenhierarchie** zugeordnet werden.

Zwischen **Rollentyp** und **Rolle** besteht eine 1:n Beziehung. Einem Rollentyp können beliebig viele Rollen zugeordnet werden. Eine Rolle bezieht sich auf genau einen Rollentyp. Eine Rolle kann null bis beliebig viele Zugriffsrechte (**Rechte)** im Beziehungsobjekttyp **Rollenrechte** bündeln. Ein Zugriffsrecht (**Rechte)** kann null bis beliebig vielen Rollen zugeordnet werden. Zwischen **Rolle** und **Rechte** besteht eine n:m Beziehung.

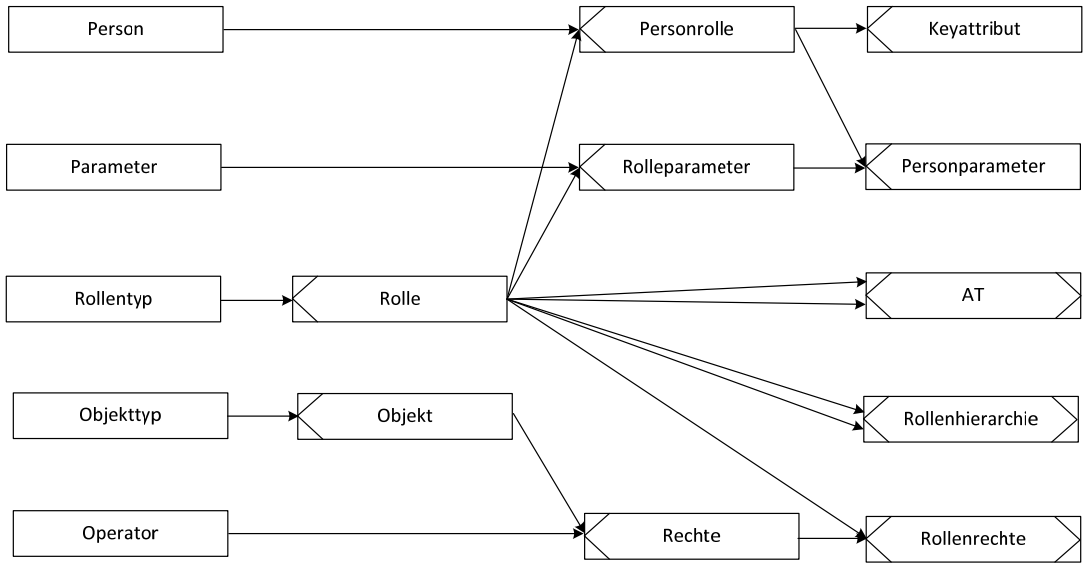

Abb. A-2 Datenschema der Autorisierung

Die Subjektzuordnung ist eine m:n Beziehung zwischen **Person** und **Rolle**. Einer **Person** können null bis beliebig viele Rollen (**Rolle**) im Beziehungsobjekttyp **Personrolle** zugeordnet sein. Eine Rolle kann sich ebenso auf null bis beliebig vielen Personen beziehen. Einem Eintrag in **Personrolle** können null bis beliebig viele Schlüsselattribute in **Keyattribut** zugeordnet werden. Jeder Eintrag in **Keyattribut** bezieht sich auf genau einen Eintrag in **Personrolle**. Es besteht dabei eine 1:n Beziehung.

Für eine **Rolle** sind null bis beliebig viele **Parameter** im Gegenstands-Beziehungsobjekttyp **Rolleparameter** hinterlegt. Zwischen **Rolle** und **Parameter** besteht eine n:m-Beziehung. Einem Eintrag in **Personrolle** können null bis beliebig viele Schlüsselattribute für einen **Parameter** im Gegenstands-Beziehungsobjekttyp

Personparameter hinterlegt werden. Es besteht eine 1:n Beziehung zwischen **Personrolle** und **Personparameter** sowie zwischen **Rolleparameter** und **Personparameter**.

A.2.2 Datenschema der Protokollierung

In **Abb. A-3** ist das Datenschema der Protokollierung dargestellt. Beschrieben werden die Entitäten: **Sitzung**, **Sitzungsrollen**, **Se_Ro_Keyatt** und **Protokoll**. Die Entitäten: Person, Rollentyp, Rolle, Objekttyp, Objekt und Operator beschreibt Kapitel A.2.1 und werden hier aufgenommen, um die Fremdschlüsselbeziehung aufzuzeigen. Diese sind durch die gestrichelten Linien gekennzeichnet.

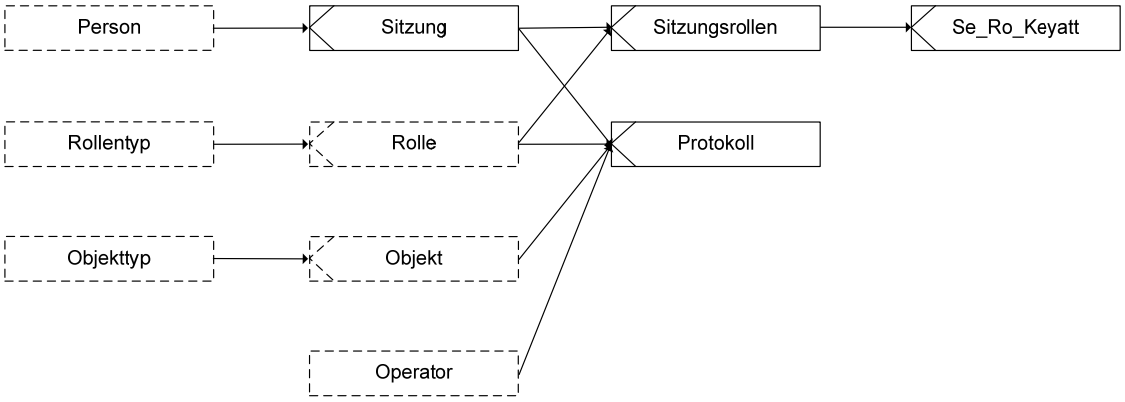

Abb. A-3 Datenschema der Protokollierung

Das Datenschema der Protokollierung umfasst die folgenden Tabellen:

- Die Tabelle **Sitzung** enthält die Sitzungsinformationen über das Subjekt, das sich authentifiziert hat, den Startzeitpunkt der Sitzung und nach erfolgtem Abmelden den Endzeitpunkt der Sitzung.
- In der Tabelle **Sitzungsrollen** werden alle Rollen gespeichert, die dem authentifizierten Subjekt direkt zugeordnet sind. Zudem wird beim Anwählen einer Anwendung implizit auch die dazugehörige Rolle aktiviert und als aktiv gekennzeichnet.
- In der Tabelle **Se_Ro_Keyatt** werden alle Schlüsselattribute pro Session, Rolle und Datenobjekt für die Personalisierung gespeichert.
- In der Tabelle **Protokoll** wird jede erfolgreiche und erfolglose Autorisierung mit den Informationen, welches Subjekt, welches Objekt mit welchem Operator aufgerufen hat, gespeichert.

Sitzung, **Sitzungsrollen**, **Se_Ro_Keyatt** protokollieren eine erfolgreiche Authentifizierung. **Protokoll** protokolliert die einzelnen Zugriffe eines Subjektes.

Einer **Person** können null bis beliebig viele Sitzungen (**Sitzung**) zugeordnet sein, jede Sitzung bezieht sich auf genau eine Person. Jeder Sitzung sind null bis beliebig viele Rollen über den Gegenstands-Beziehungsobjekttyp Sitzungsrollen zugeordnet. Eine Sitzung kann sich auf beliebig viele Rollen beziehen und umgekehrt eine Rolle kann in beliebig vielen Sitzungen vorkommen. Jeder **Sitzungsrolle** können null bis beliebig viele Schlüsselattribute in **Se_Ro_Keyatt** zugeordnet werden. Zwischen diesen Tabellen besteht eine 1:n Beziehung.

Einer **Sitzung** können null bis beliebig viele Einträge für eine Beziehung zwischen einer aufgerufenen **Rolle** und dem **Objekt** mit seinem zugehörigen **Operator** im Gegenstands-Beziehungsobjekttyp **Protokoll** zugeordnet sein.

Danksagung

Während meiner Tätigkeit als Projektmitarbeiterin und Geschäftsführerin der Abteilung 1 am Wissenschaftlichen Institut für Hochschulsoftware der Universität Bamberg (ihb) entstand die vorliegende Arbeit.

Für die persönliche Betreuung, wertvolle Unterstützung und konstruktiven Hinweise bei der Erstellung der Arbeit möchte ich mich vor allem bei meinem Doktorvater Herrn Prof. Dr. Elmar J. Sinz bedanken. Für die Übernahme des Zweitgutachten danke ich Herrn Prof. Dr. Sven Overhage. Bei Frau Prof. Dr. Ute Schmid bedanke ich mich für die Begleitung der Arbeit als Mitglied der Promotionskommission und den steten Zuspruch mein Ziel nicht aus den Augen zu verlieren.

Ein besonderer Dank gilt dem gesamten FlexNow-Team für die jahrelange, erfolgreiche und freundschaftliche Zusammenarbeit. Durch ihre Unterstützung hatte ich die Möglichkeit mich dieser Arbeit zu widmen. Besonders bedanken möchte ich mich bei Florian Bader für die zahlreichen fachlichen Diskussionen und konstruktiven Impulse sowie bei Annette März-Löwenhaupt für das Korrekturlesen der Arbeit.

Ohne den familiären Rückhalt, den ich von allen Familienmitgliedern kontinuierlich erfahren habe sowie deren konstante Ermutigung, wäre diese Arbeit nicht möglich gewesen. Dafür möchte ich mich auch ganz herzlich bedanken.

Bamberg, November 2015

Gerlinde Fischer