IT-Sicherheit

Kapitel 4: Authentifizierung und Autorisierung

Teil 2

Prof. Dr. Reiner Hüttl, TH Rosenheim, © 2021, 15.03.2021

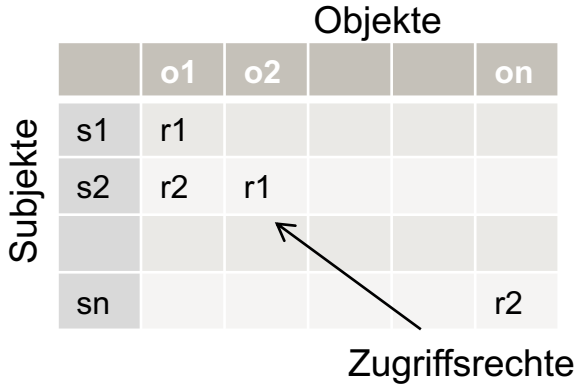
Zugriffskontrolle - Access Control - Authorization

- **Discretionary-Access-Control (DAC)**
  - Benutzerbestimmbare Zugriffskontrolle
  - Jeder Besitzer kann Rechte auf seine Objekte an andere Benutzer übertragen
  - Die Rechtevergabe ist dezentral gesteuert

- **Mandatory Access Control (MAC)**
  - Systembestimmte (regelbasierte) Festlegung von Sicherheitseigenschaften
  - Benutzerdefinierte Rechte werden durch systembestimmte überschrieben (dominiert)
  - Zusätzlich werden Sicherheitsklassen und globale Regelungen eingeführt
  - Betriebssysteme oder Anwendungen müssen spezielle Maßnahmen und Dienste bereitstellen, um MAC-Policies durchzusetzen

Realisierung der Zugriffkontrolle über Zugriffsmatrix

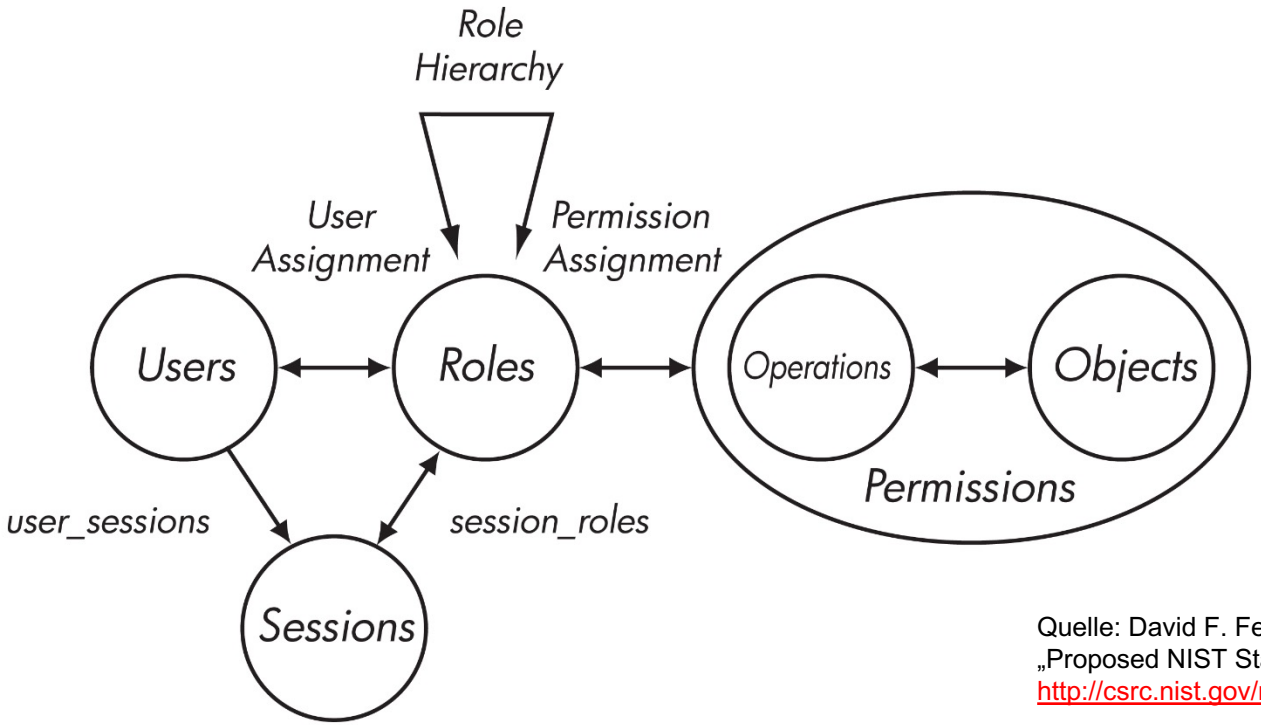
- Zugriffskontrolle wird oft mit einer (dünnbesetzten) Zugriffsmatrix realisiert, die in zwei Dimensionen betrachtet werden kann



- Zugriffskontrollliste – **Access Control Lists (ACL)**
  - Objektbezogene Sichtweise, eine Liste per zu schützenden Objekt
  - Definieren die Zugriffsrechte von Subjekten auf Objekte
  - Vorteil: einfache Verwaltung und Rechterücknahme
  - Nachteil: z.T. ineffizient bei vielen Subjekten
- Zugriffsausweise – **Capabilities (Berechtigungen)**
  - Subjektbezogene Sichtweise
  - Unfälschbare Tickets, die den Inhaber zum Zugriff auf ein Objekt berechtigen
  - Vorteil: Flexibel, dezentral, geeignet für Delegation
  - Nachteil: Rechterücknahme aufwändig

Rollenbasierte Zugriffskontrolle (RBAC)

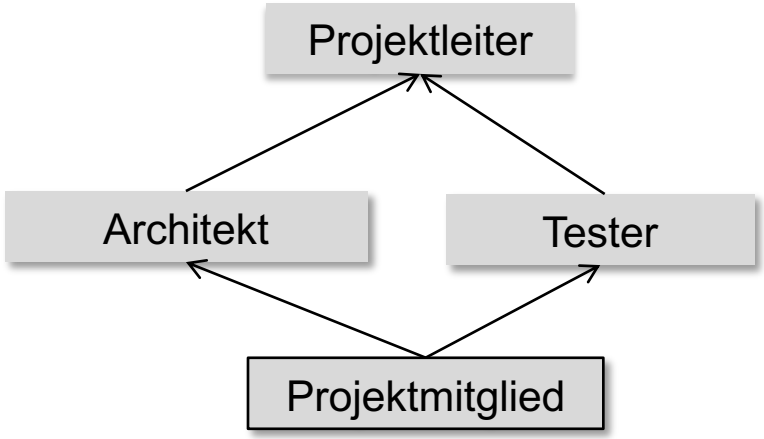
- **Rolle-Based-Access**-Control-Pattern
  - Rollen werden Berechtigungen für Objekte zugewiesen (**pr = permission to role**)
  - Subjekte werden Rollen zugewiesen (**sr = subject to role**)



Quelle: David F. Ferraiolo, Ravi Sandhu, Serban Gavrila, R. Richard Kuhn, „Proposed NIST Standard for Role-Based Access Control“, 2001
http://csrc.nist.gov/rbac/rbacSTD-ACM.pdf

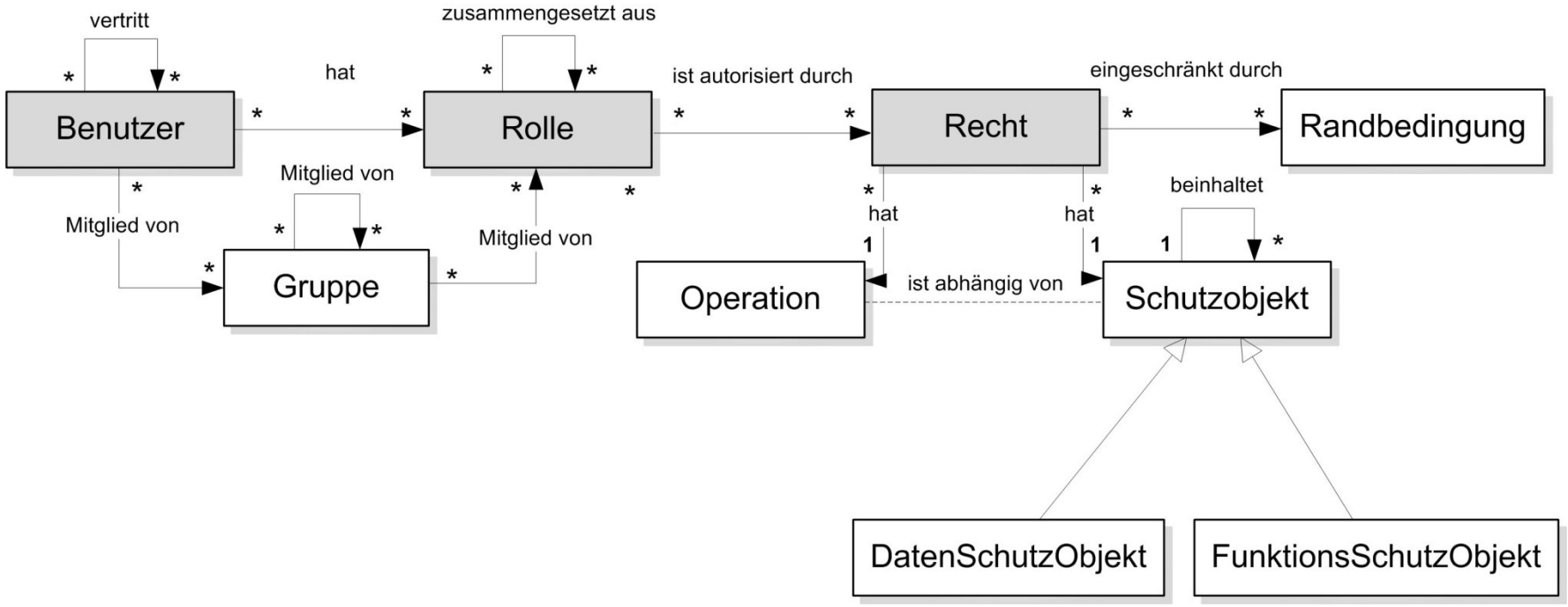
Bestandteile eines RBAC-Modells

- Sessions
  - Eine Session bedeutet ein Subjekt ist aktiv in einer Rolle
  - Ein Subjekt darf nur in Rollen aktiv sein in denen er Mitglied ist
  - Ein Subjekt besitzt nur die Rechte seiner aktiven Rolle
- Rollenhierarchie
  - Ziel : Nachbildung von Organisationsstrukturen
  - Definition einer partiellen Ordnung auf Rollen
    $R_i, R_j \in Role, falls\ R_i \leq R_j$
    dann besitzt R_i alle Rechte von R_j



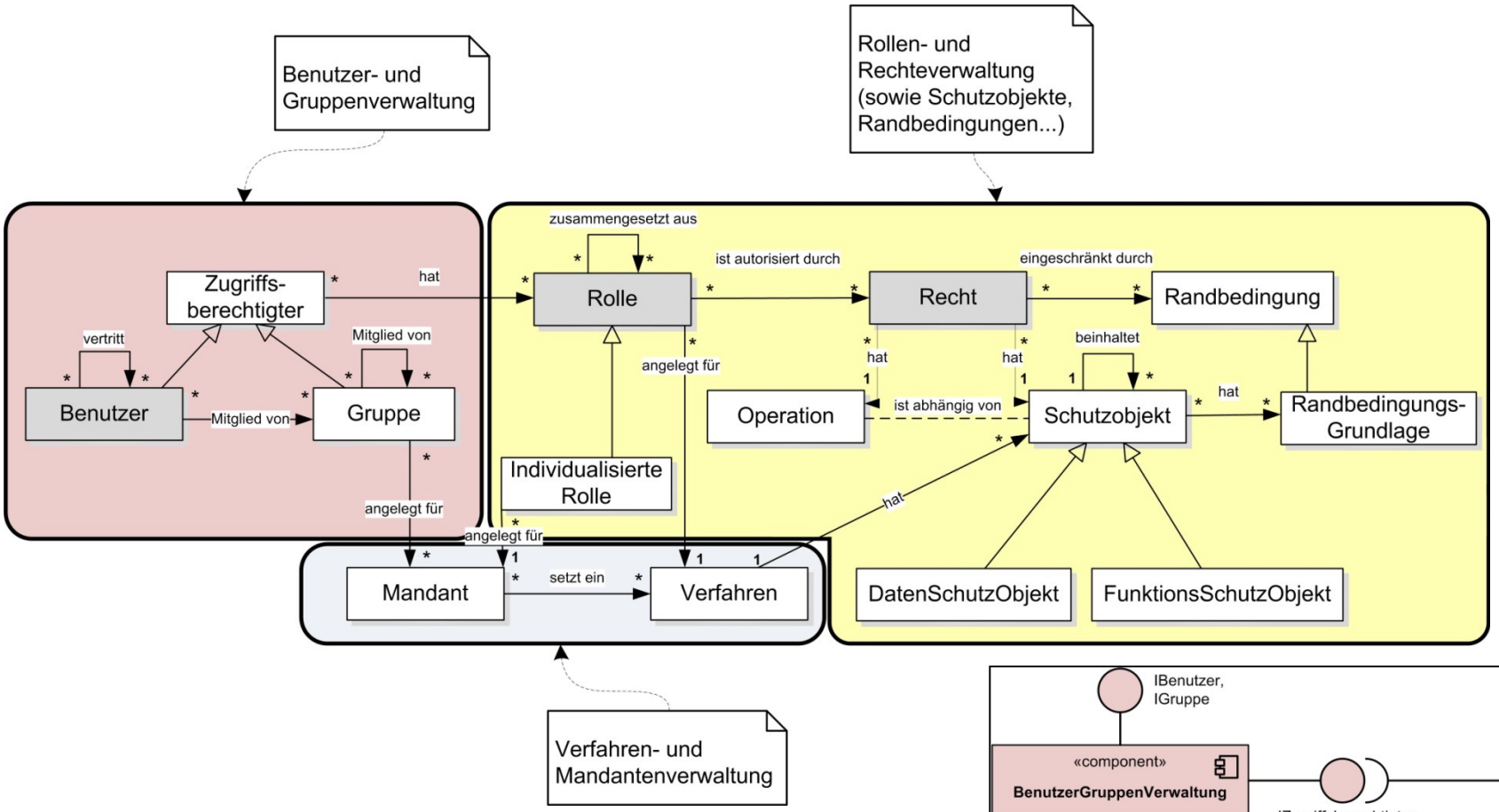
- Statische Aufgabetrennung: wechselseitiger Ausschluss von Rollenmitgliedschaften

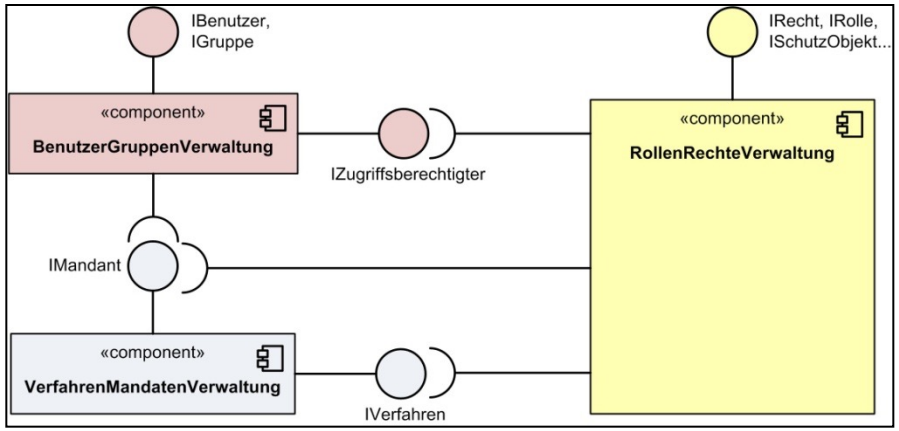
Datenmodell für eine Autorisierungskomponente



[Diplomarbeit Sebastian Keller, WS2008, FH Rosenheim]

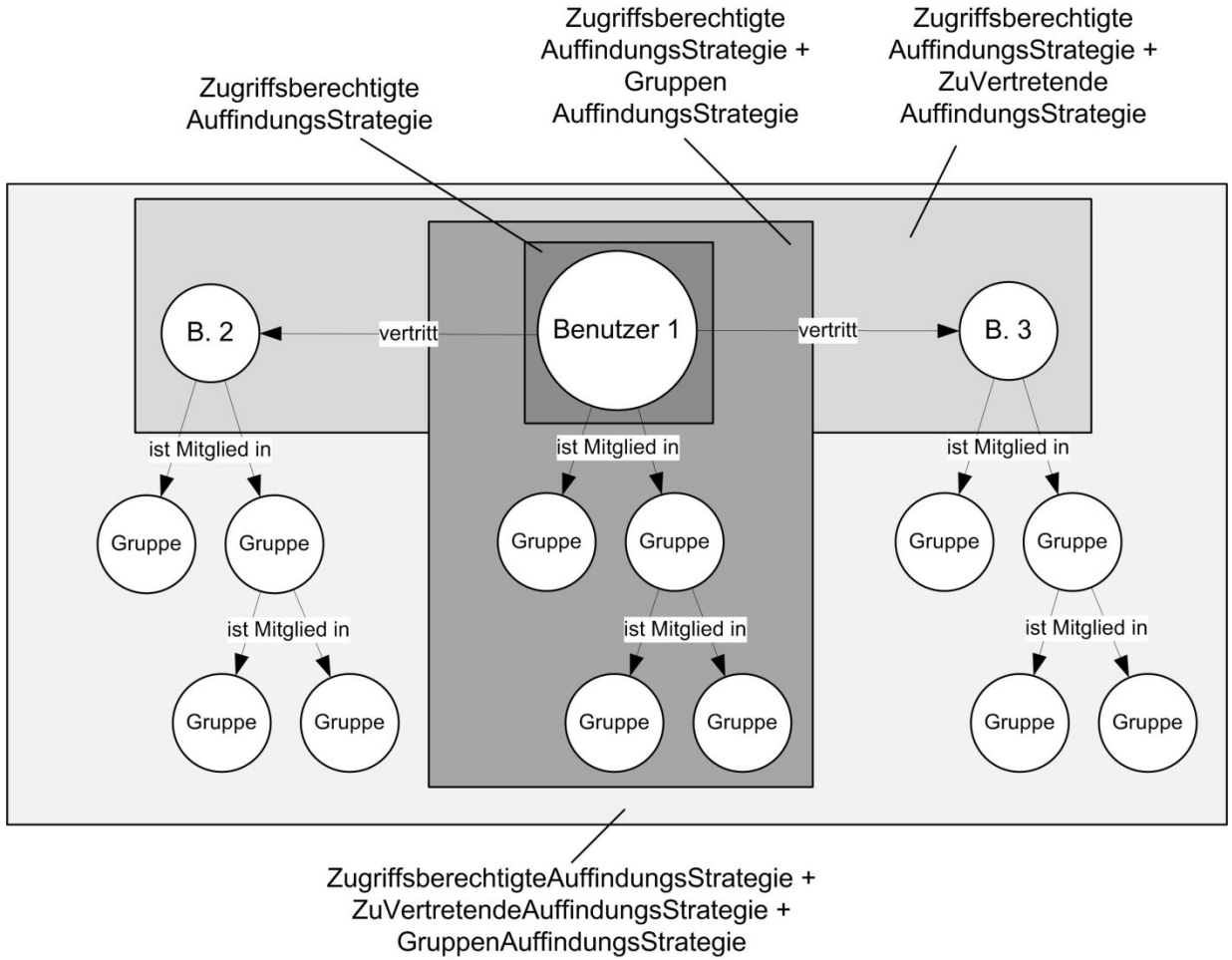
Entwurf der Komponente aus dem Datenmodell





[Diplomarbeit Sebastian Keller, WS2008, FH Rosenheim]

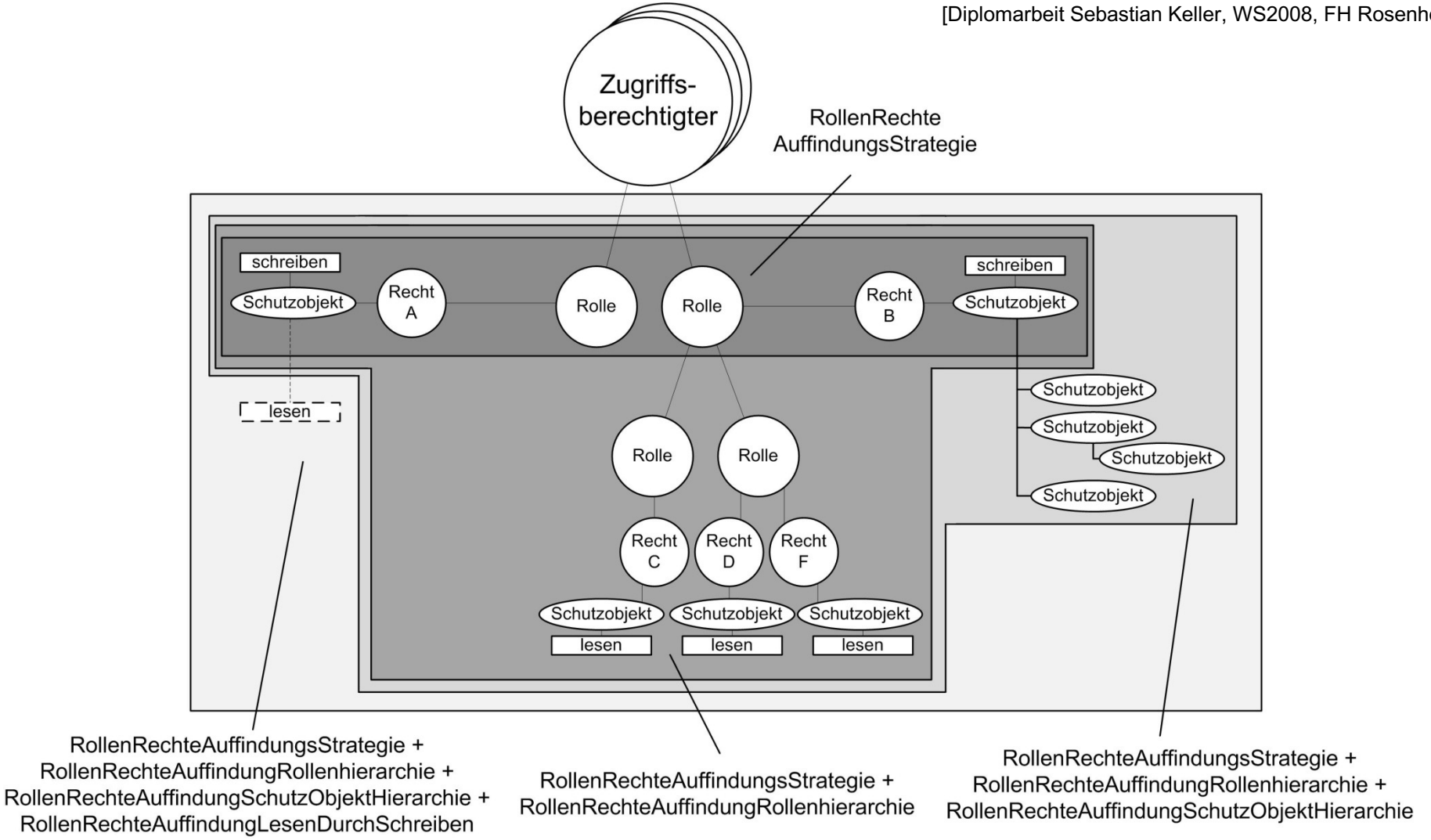
Auswertung der Zugriffsrechte mit Gruppen und Vertreter



[Diplomarbeit Sebastian Keller, WS2008, FH Rosenheim]

Auswertung der Zugriffsrechte mit Rollen-und Schutzobjekt-Hierarchie

[Diplomarbeit Sebastian Keller, WS2008, FH Rosenheim]



Bewertung RBAC-Modell

- Rollenkonzepte sind sehr flexibel verwendbar, aufgabenorientiert, administrierbar und skalieren gut
- Sie ermöglichen ein direktes Nachbilden bekannter Organisations- und Rechtestrukturen in Unternehmen: gute Basis für ID-Management
- Intuitive und relativ einfache Abbildung der Rollen auf Geschäftsprozesse (Workflows): **Need-to-know-**Rechtvergabe
- Änderungen von ***pr*** *selten;*
  dagegen aber u.U. Änderung der Rollenmitgliedschaften ***sr*** *häufig;*
- Einfache und effiziente Rechte-Verwaltung, automatischer Rechteentzug bei Mitgliedschafts-Ende
- Gefahr: Rollen werden missbraucht um Berechtigungen darzustellen, was zu einer explodierenden Anzahl von Rollen führen kann

Rule-Based Access Control (RuBAC)

- Zugriffskontrolle auf Basis von Regeln
- Typische Einsatzgebiete:
  Firewalls, Router
- Bei Benutzerrechten kann es bei MAC eingesetzt werden
  - Regeln beschreiben Situationen in denen ein Subjekt auf ein Objekt zugreifen kann
  - RuBAC werden schnell sehr komplex
- Regeln können mit Policies beschrieben werden:
  Policy-Based-Access Control PBAC

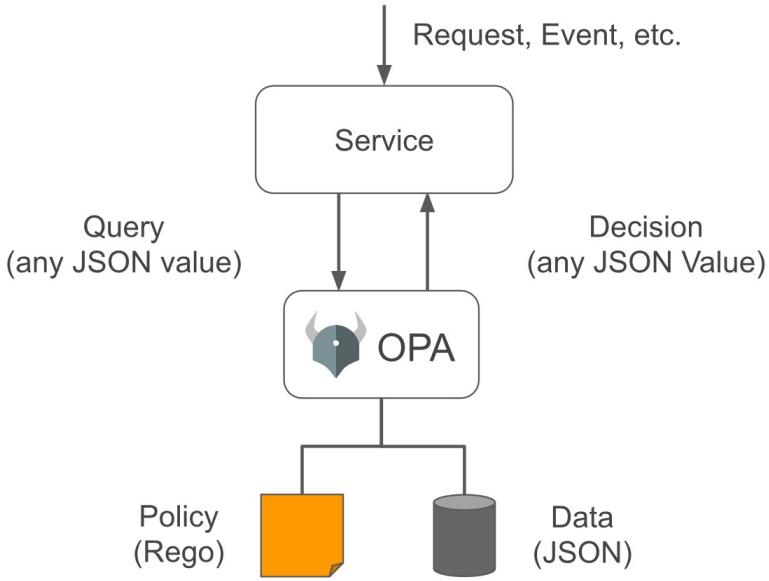
```
# Allow users to get their own salaries.
allow {
  input.method = "GET"
  input.path = ["finance", "salary", username]
  input.user == username
}

# Allow managers to get their subordinates' salaries.
allow {
  input.method = "GET"
  input.path = ["finance", "salary", username]
  subordinates[input.user][_] == username
}
```

Example of an Open Policy Agent Policy in Policy language Rego („ray-go“)

Open Policy Agent

Open Policy Agent OPA ermöglicht Rule Based Access Control in Cloud Umgebungen

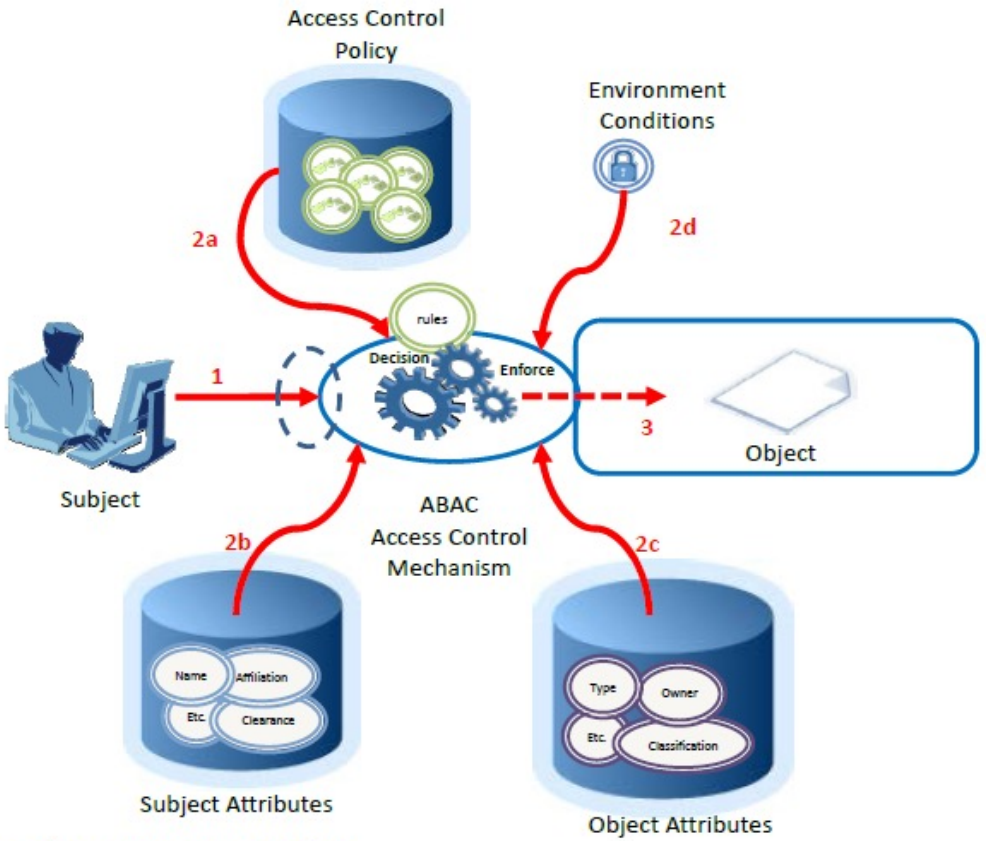
OPA enables fine-grained policy-based control in cloud native environments



OPA generates policy decisions by evaluating the query input and against policies and data

Quelle: https://www.openpolicyagent.org/

Attibute Based Access Control (ABAC)



1. Subject requests access to object
2. Access Control Mechanism evaluates a) Rules, b) Subject Attributes, c) Object Attributes, and d) Environment Conditions to compute a decision
3. Subject is given access to object if authorized

- OpenID Connect ermöglicht Zugriffskontrolle auf Basis von Attributen (claims)
- ABAC ist merkmalbasiert
- XACML eXtensible Access Control Markup Language: attributsbasierte Zugriffskontroll-Policy Sprache
- Einsatzgebiet:
  - API-Gateway bei Micro Services
  - Zugriff auf Big Data Systeme

Quelle: NIST http://nvlpubs.nist.gov/nistpubs/specialpublications/NIST.sp.800-162.pdf

Access Control Patterns

- **Least privilege**: a subject should be given only those privileges needed for it to complete its tasks, raises system stability and security
- **Need to Know**: user gets access only if it‘s necessary to conduct its duties
- **Separation of Duty**: more than one user is required to complete a task, increases protection from fraud and errors, control against insider attacks
- **Separation of Concerns**: separate a computer program into distinct sections
- **Open Policy**: everything is allowed which is not forbidden
- **Closed Policy**: only explicit authorized access is allowed
- **Dual Control:** Four eyes principle, two or more separate entities are necessary to access sensitive functions or information

Zusammenfassung Authentifizierung und Autorisierung

- Es gibt viele verschiedene Varianten zur Authentifizierung
- Sie unterscheiden sich in Sicherheit, Mobilität, Kosten und Bequemlichkeit
- Für SSO gibt es verschiedene Standards und Technologien (Kerberos, OAuth, OpenID, SAML)
- Die RBAC ist ein sehr flexibles Modell zur Verwaltung von Zugriffsrechten
- In modernen Cloud Umgebungen spielen auch RuBAC und ABAC eine wichtige Rolle