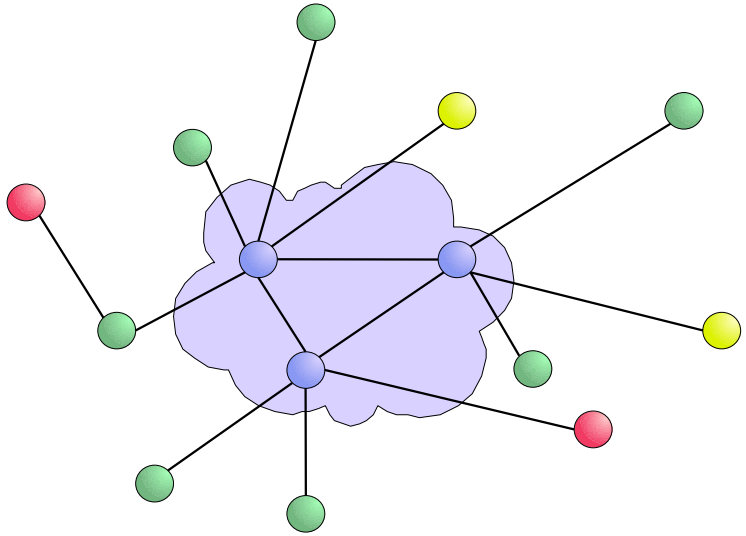
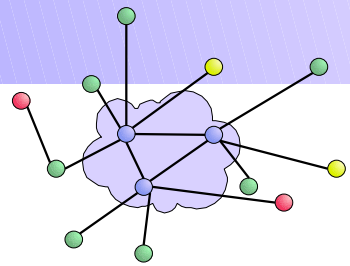


Sebastian Bächle

Sicherheit
in verlässlichen adaptiven Informationssystemen

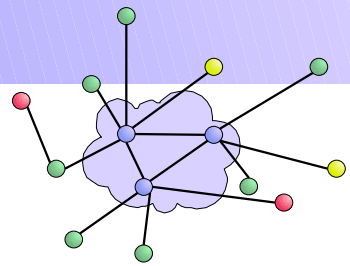
Übersicht



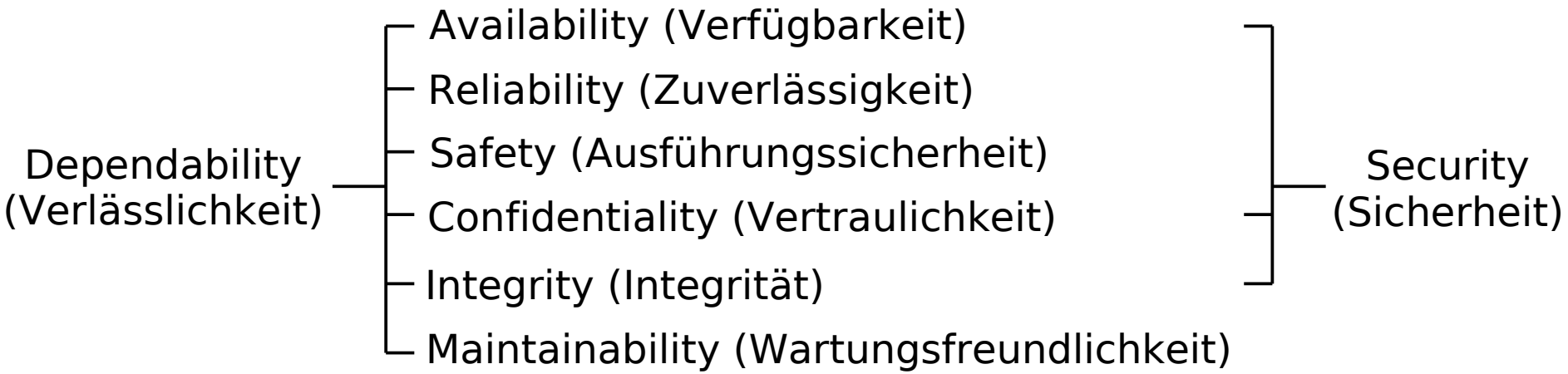
- Begriffsklärungen
- Allgemeine Sicherheitsmaßnahmen
- Entwurf sicherer Systeme
- Vertrauen in verteilten Systemen
- Intrusion Detection
- Fazit

Verlässlichkeit vs. Sicherheit

Begriffsklärungen

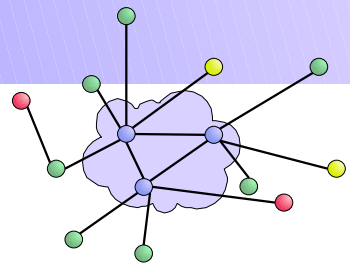


- Sicherheit ist wie Verlässlichkeit eine Kombination verschiedener Qualitäten
- Sicherheit ist eine Voraussetzung für Verlässlichkeit!



Fehler

Begriffsklärungen

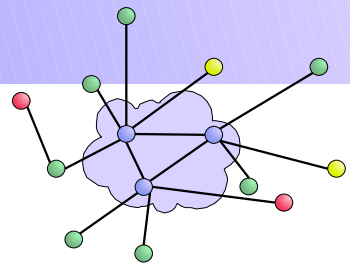


- Service Failure (Fehlfunktion): Ereignis, dass ein Dienst von seiner korrekten Durchführung abweicht
- Korrektheit bezieht sich nur auf die Spezifikation des Dienstes
- Error (Fehlerzustand): Teil des Gesamtsystemzustandes, der zu einer Fehlfunktion führt
- Fault (Fehler): Ursache eines Fehlerzustandes
- Chain of Threats:

- Ein System ist verwundbar, wenn intern ein Fehler existiert, der es einem externen Fehler ermöglicht Schaden zu verursachen

Wie erhöht man die Sicherheit?

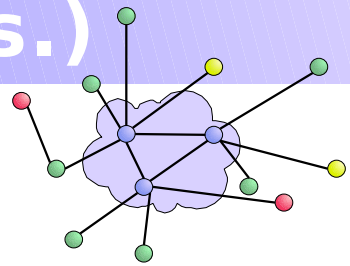
Allgemeine Sicherheitsmaßnahmen



- Zugriffskontrolle:
  - Authentifikation und Autorisierung (Kerberos, PAM, JAAS)
- Schutz der Daten:
  - Verschlüsselung (TLS/SSL, SSH)
  - temporale, logische und physische Datentrennung
- Architektur-Maßnahmen:
  - Redundanz, Design-Diversität
- Isolierung von Programmen:
  - Virtualisierung (z/VM, VMware, Xen)
  - Sandboxing (Java)

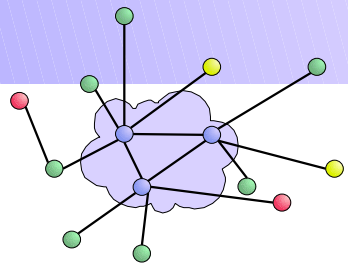
Wie erhöht man die Sicherheit? (Forts.)

Allgemeine Sicherheitsmaßnahmen



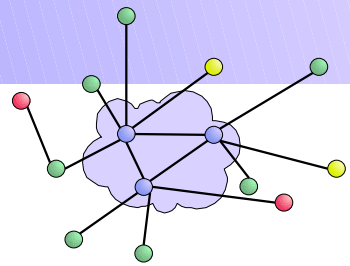
- Firewall, DMZ und Portfiltering
- Täuschungs- und Störeinrichtungen:
  - Honeypots, Tarpits
- Einbruchserkennung (Intrusion Detection)
- Recovery und Adaption
- Sicherheitswartung
- sicherheitsbewusstes Personalmanagement!
  - Aufklärung über Social Engineering

Modellierung von Sicherheit

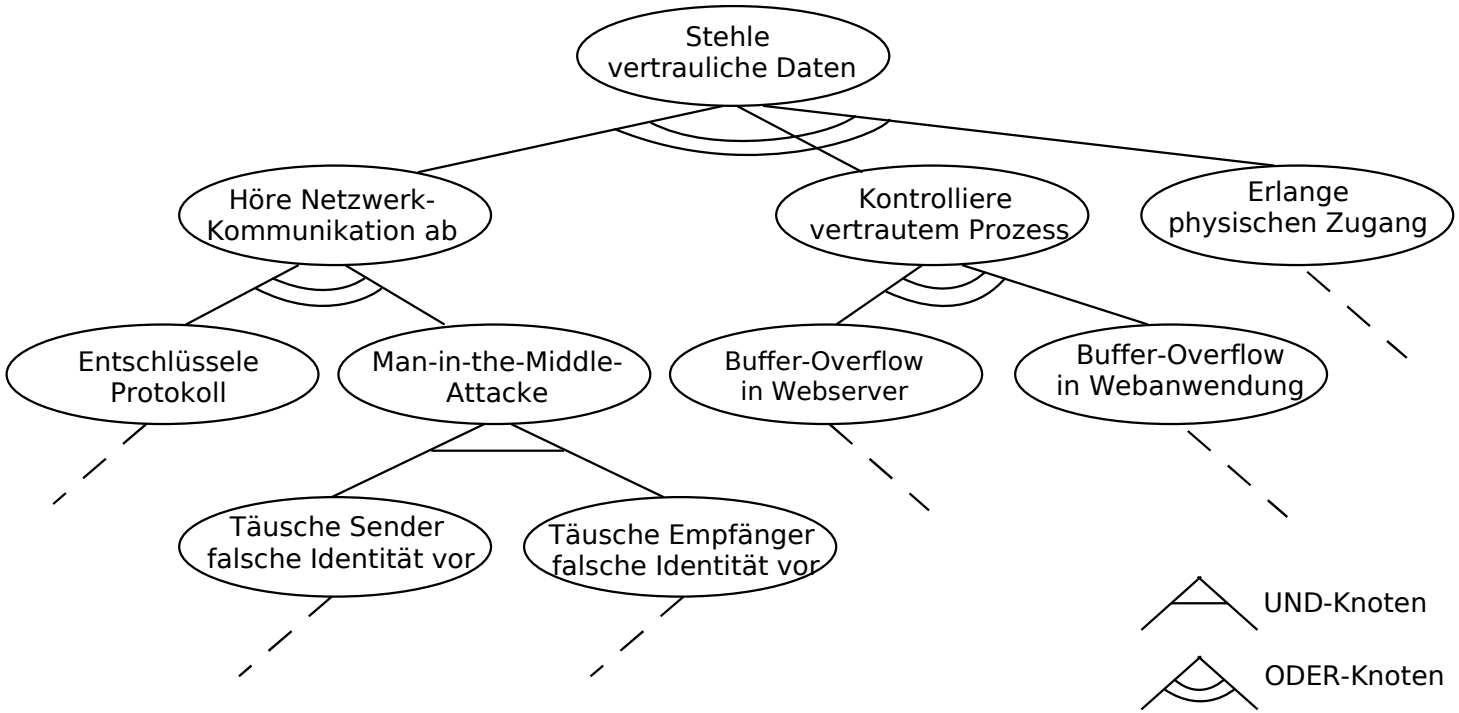


- Fragestellungen: Kann man die Sicherheit eines Systems
  - modellieren?
  - messen?
  - überprüfen?
  - mit einem anderen System vergleichen?
- Es existiert kein „Modell von Sicherheit“
- Aber: Sicherheitsmaßnahmen können spezifiziert werden!

Identifikation von Bedrohungen

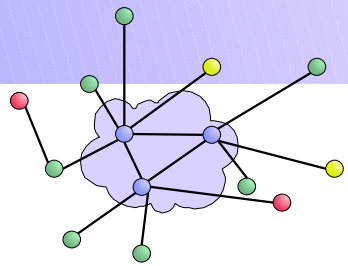


- Potentielle Gefährdungen sollten soweit wie möglich vor dem Entwurf identifiziert werden
- Wichtig für Dokumentation und Strukturierung
- Bsp. Attack Trees



UMLsec

Entwurf sicherer Systeme



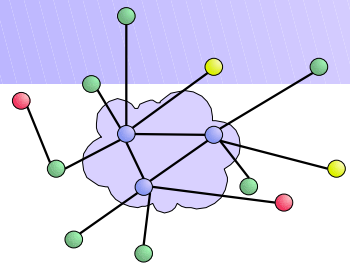
- Überprüfbarkeit von Sicherheit:

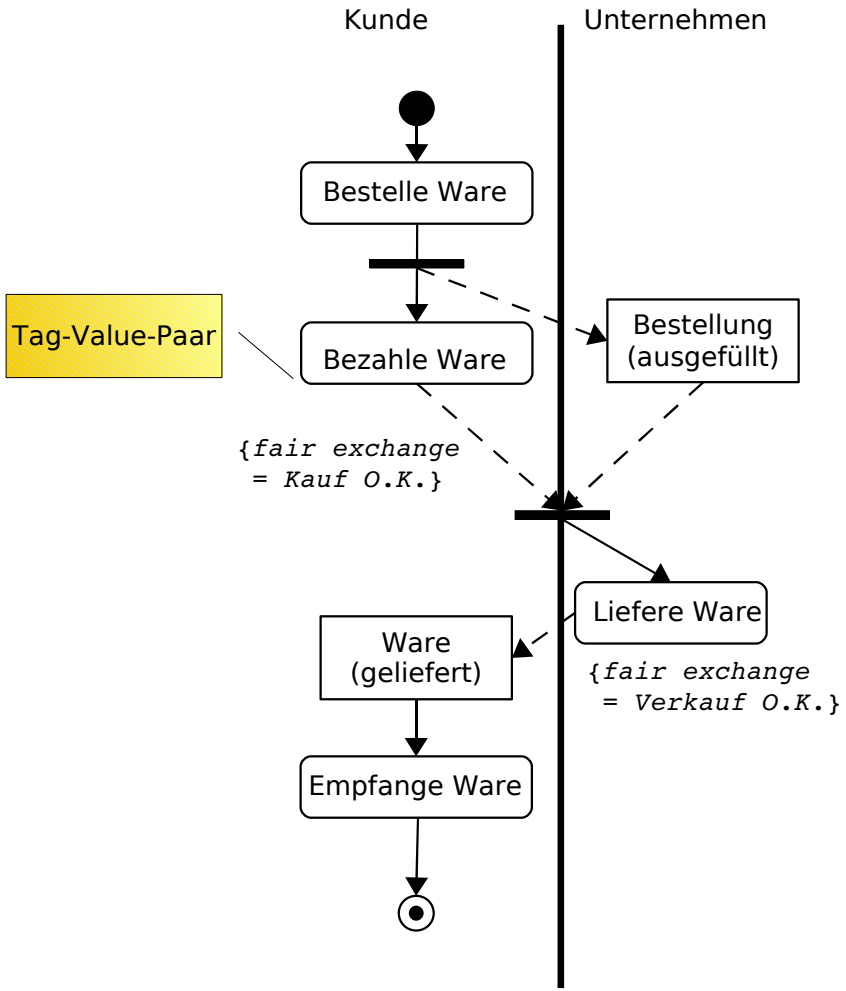
  qualitative Sicherheitsanforderung ➡ funktionale Sicherheitsanforderungen

- Modellierung mit UML + UMLsec
- UMLsec = Profil mit neuen Stereotypen und Tag-Value-Paaren zum Markieren von Diagrammelementen
- unterstützt fast alle Diagrammtypen
- Beispiele:
  - <<*encrypted*>> als Subtyp von <<*link*>>
  - Tag *secret* zum Markieren geheimer Daten

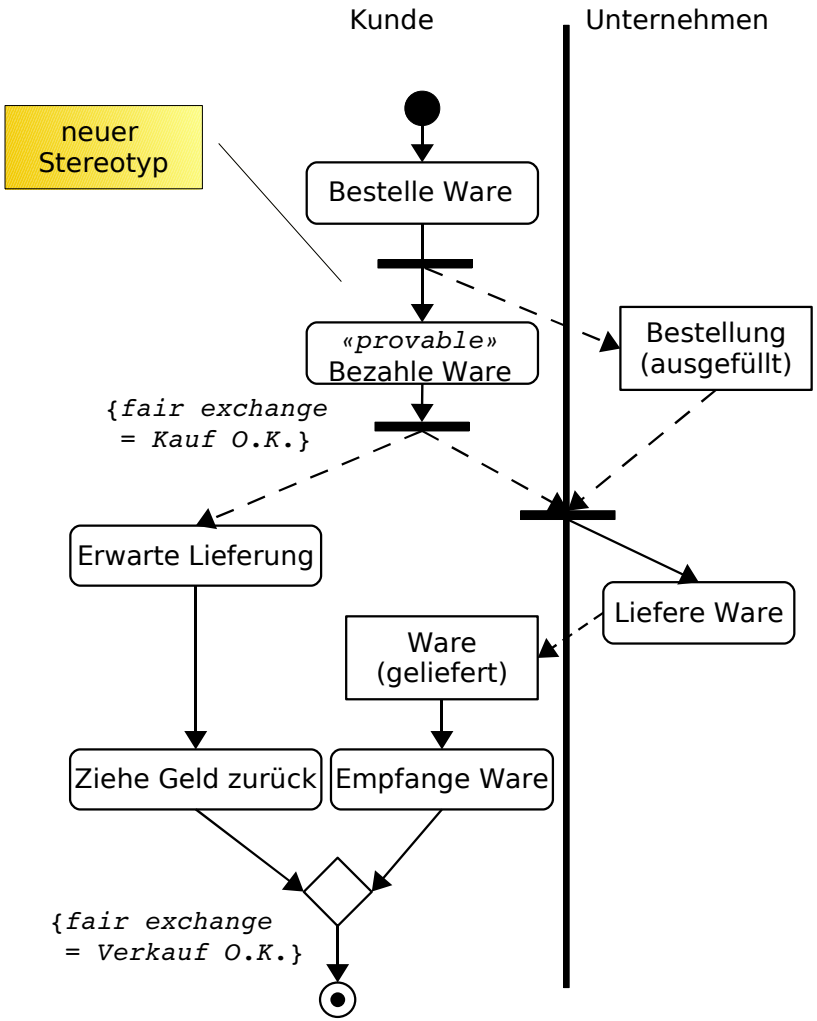
UMLsec (Forts.)

Entwurf sicherer Systeme



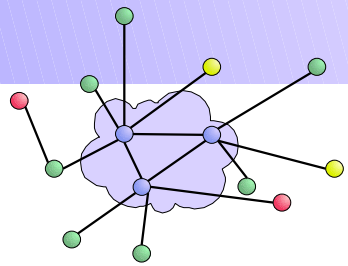


fair exchange VERLETZT



fair exchange GESICHERT

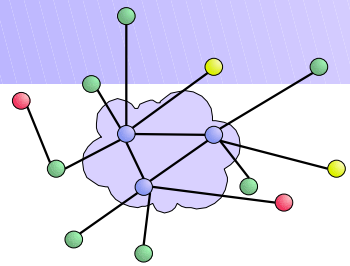
Überprüfbarkeit von Sicherheit



- Verifikation eines Systementwurfs bedeutet nicht, dass das System sicher ist!
- Sicherheit kann nur bis Systemgrenze modelliert werden
- Ein Systementwurf beruht immer auf Trust Assumptions:
  - vertrauenswürdiger Administrator
  - Mitarbeiter gehen sorgfältig mit ihren Passwörtern um
  - Compiler erzeugt korrekten Code
- Sicherheit lässt sich nicht
  - verifizieren
  - garantieren

Bedeutung von Vertrauen für DAIS

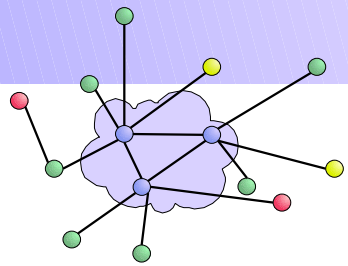
Vertrauen in verteilten Systemen



- Anforderungen an ein DAIS:
  - Es soll möglichst vielen Benutzern zur Verfügung stehen
  - Die Interaktion mit neuen Partnern soll schnell und mit geringem Aufwand ermöglicht werden
  - Das System soll verlässlich arbeiten
  - Das System soll sicher sein
- Lösung: Entscheide Interaktionsrisiko durch Vertrauen!
  - Aufbau von Vertrauensmodellen zur Authentifikation und Autorisierung von Parteien
  - Verwendung von Public-Key-Infrastrukturen (PKI)

Definition: Vertrauen

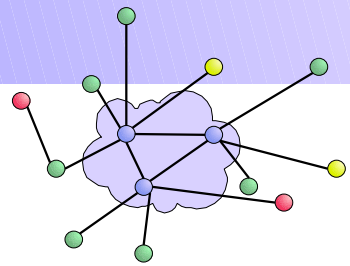
Vertrauen in verteilten Systemen



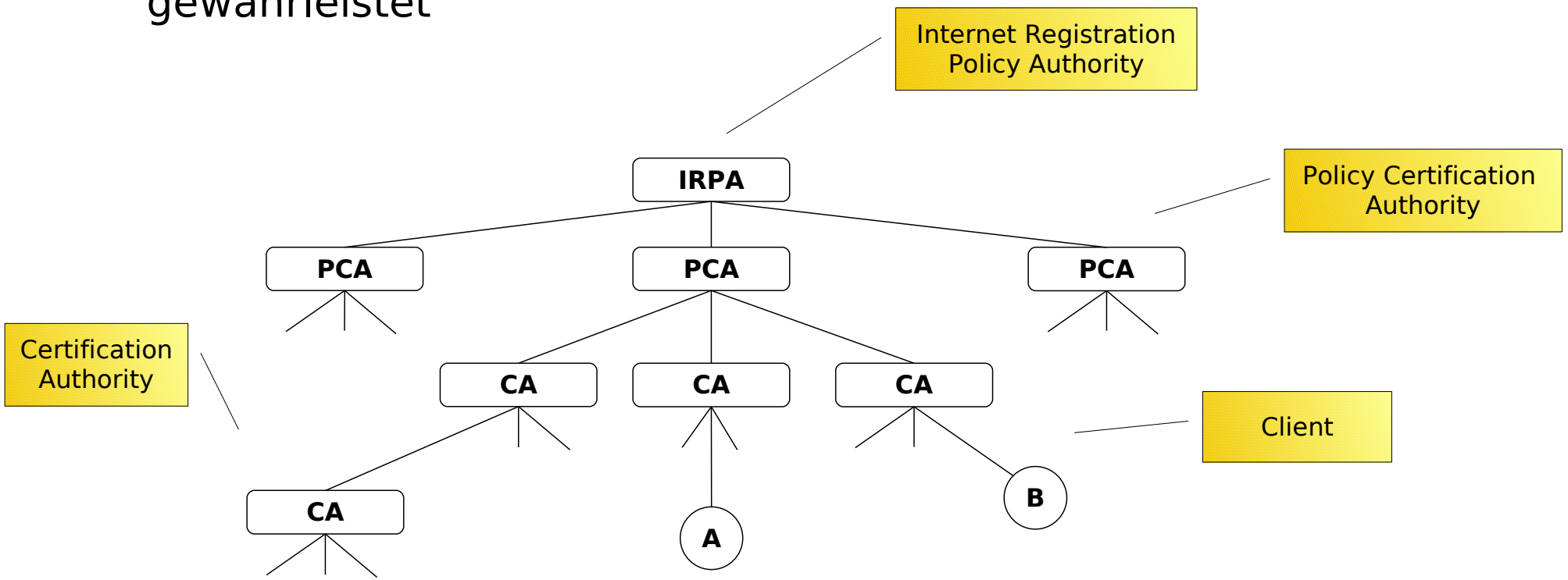
- Vertrauen ist
  - eine gerichtete binäre Relation
  - die Entscheidung des Trustor, sich auf die Leistungen des Trustee zu verlassen
  - eine akzeptierte Abhängigkeit
  - notwendig für die Interaktion mit Anderen

Hierarchische Vertrauensmodelle

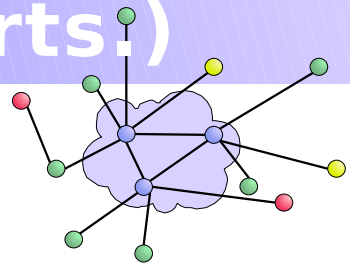
Vertrauen in verteilten Systemen



- Trust Center Hierarchy
- Identitätsnachweis mit Hilfe signierter Zertifikate (X.509)
- Identität einer Partei durch höhere Vertrauensinstanzen gewährleistet



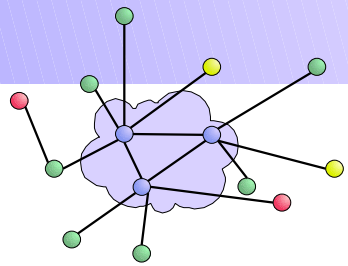
Hierarchische Vertrauensmodelle (Forts.)



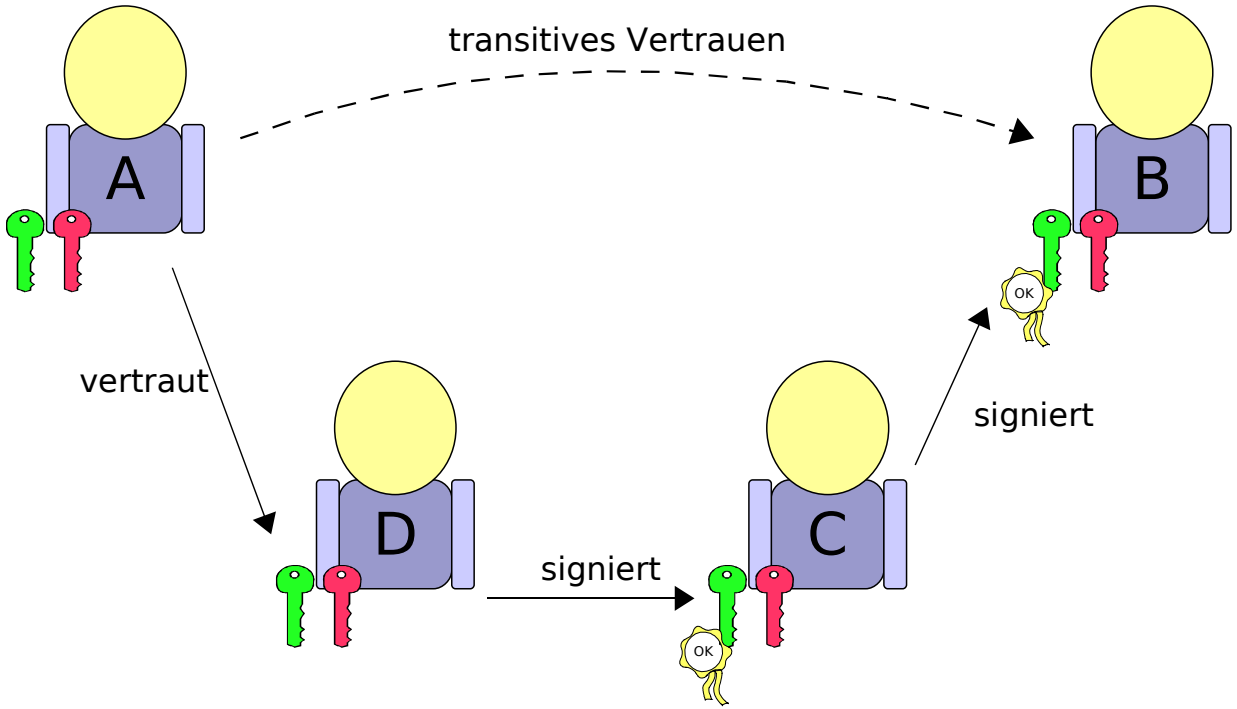
- Vorteile von Trust-Center-Hierarchien:
  - ⊕ Verfolgbarkeit des Vertrauensursprungs
  - ⊕ potentiell hohe Vertrauenswürdigkeit
  - ⊕ Standard in der Gesetzgebung und im Internet
- Nachteile:
  - ⊖ Single Point of Failure!
  - ⊖ Vertrauen beruht auf der Aussage einer einzigen Instanz

Dezentrale Vertrauensmodelle

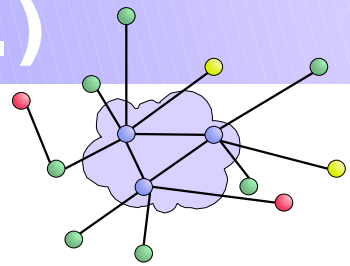
Vertrauen in verteilten Systemen



- „Web of Trust“
- Jede Partei ist eine Certification Authority!
- Schlüsselserver zum Austausch öffentlicher Schlüssel und deren Bestätigung
- direkter Schlüsselaustausch oder Vertrauen in das Netz



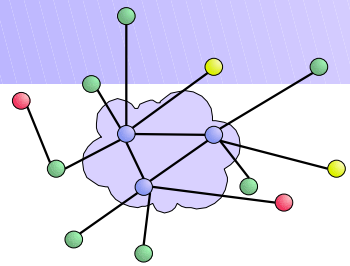
Dezentrale Vertrauensmodelle (Forts.)



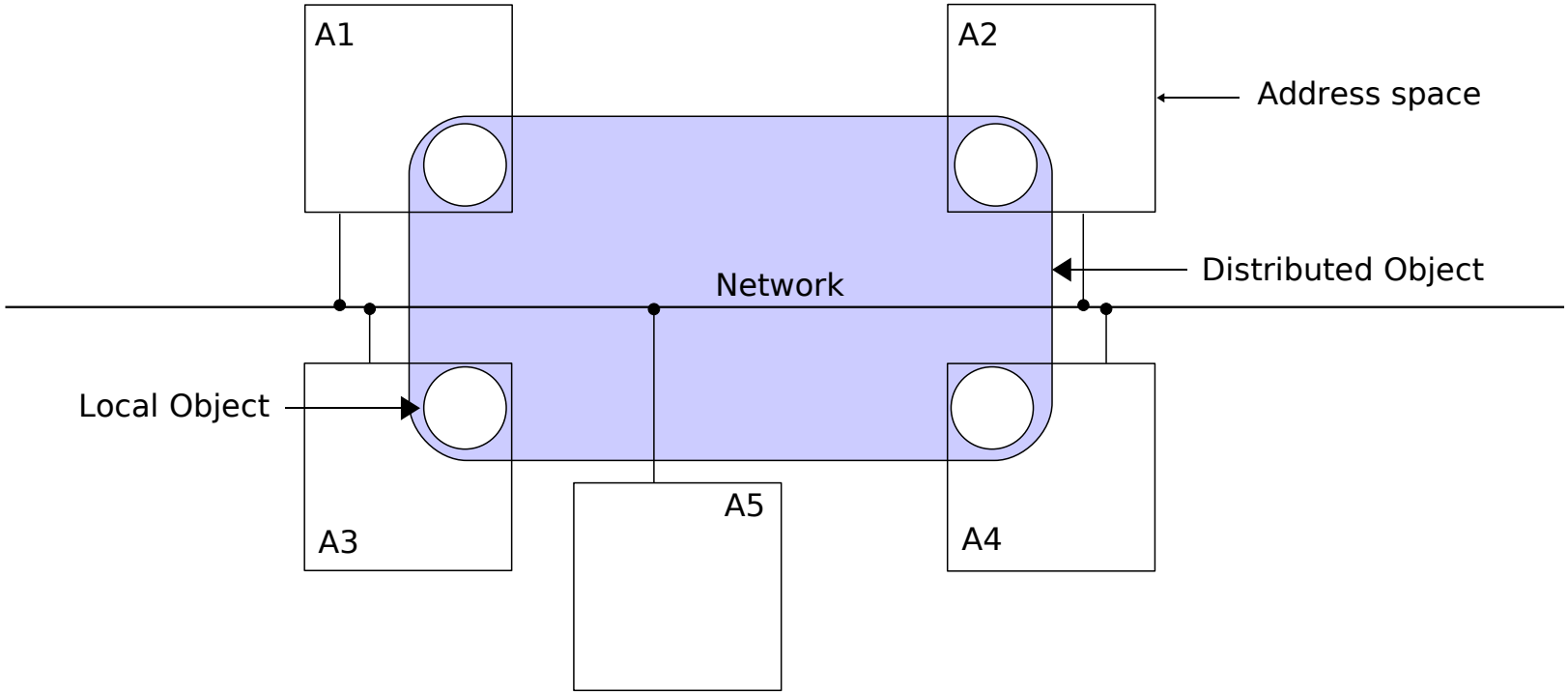
- Vorteile eines Web of Trust:
  - ⊕ dynamisch
  - ⊕ kein Single Point of Failure
  - ⊕ Ausfallsicherheit
  - ⊕ gewinnt mit stärkerer Nutzung an Sicherheit
- Nachteile:
  - ⊖ potentiell geringere Sicherheit als hierarchische Struktur
  - ⊖ kompromittierte Schlüsselserver
  - ⊖ Rechtssicherheit

Beispiel: Globe

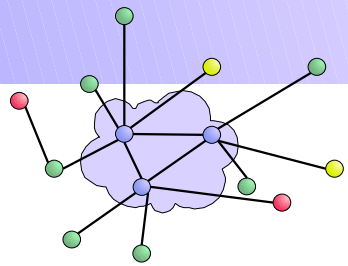
Vertrauen in verteilten Systemen

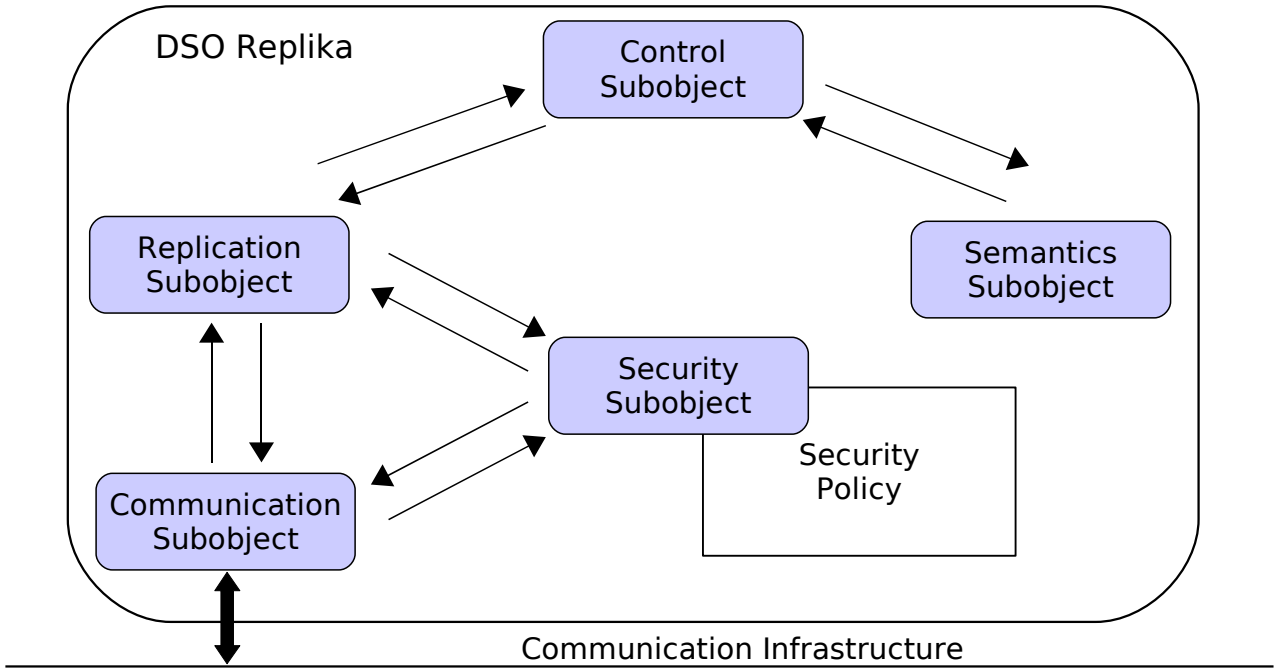


- Globe ist eine Middleware für verteilte Objekte
- Zentrales Konzept: Distributed Shared Objects (DSO)
- Logisches DSO besteht aus mehreren Local Objects
- Ziel: Replikas dynamisch so nahe wie möglich beim Benutzer platzieren



Aufbau einer Replika

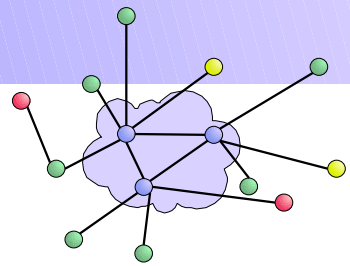




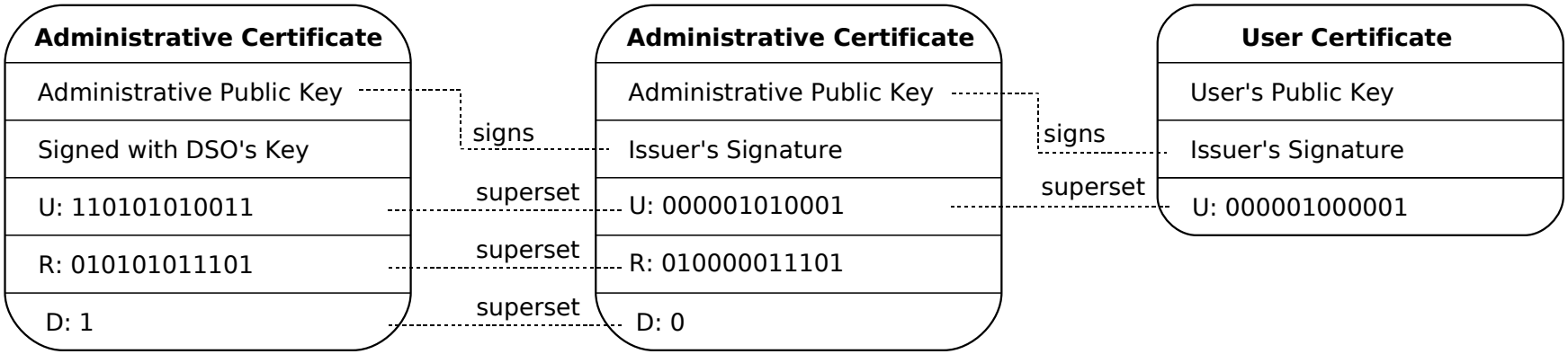
- Local Objects, die einen Teil des Zustandes des DSO speichern heißen Replika
- Globe Object Server (GOS) stellen die Laufzeitumgebung
- Zugriff über vom Benutzer erzeugte Local Objects (User Proxy)
- Security Subobject: deklarative Sicherheit durch Security Policy

Das Globe-Vertrauensmodell

Vertrauen in verteilten Systemen



- Jedes DSO, jede Replika und jeder Benutzer besitzen ein Paar aus öffentlichem und privatem Schlüssel
- Administrations-, Replika- und Benutzerzertifikate bestimmen die Rechte einer Replika bzw. eines Nutzers
- Zertifikatketten spannen den Vertrauensraum auf



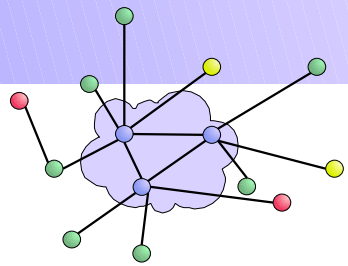
U: Bitmap with user rights

R: Bitmap with replica rights

D: Delegation bit

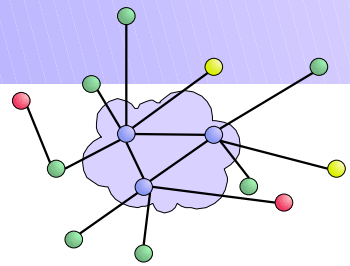
Das Globe-Sicherheitskonzept

Vertrauen in verteilten Systemen



- Sichere Bindungen zwischen
  - DSO und öffentlichen Schlüssel: ID ist Teil des Schlüssels
  - Replika und DSO: Zertifikatketten
  - DSO und repräsentiertem Geschäftsobjekt: z.B. Verbindung zum DSO über sichere HTTP-Verbindung erfragen
- Sichere Methodenaufrufe
  - spezifiziert durch Zertifikate
- Plattformsicherheit
  - Schutz des Host:
    - Sandboxing, Administrationszertifikate
  - Schutz der Replika:
    - Zertifikate, State Signing

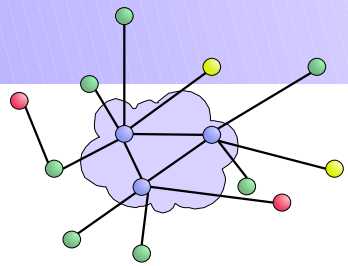
Intrusion Detection



- Grundannahme:

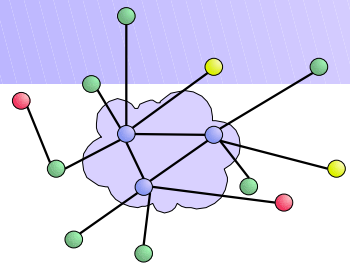
  „Das Ausnutzen von Systemschwachstellen bedingt eine anormale Nutzung, weshalb Verletzungen der Systemsicherheit anhand dieser anormalen Nutzungsmuster erkannt werden können.“
- IDS laufen parallel zur eigentlichen Funktionalität des Systems
- Intrustion-Detection-Systeme (IDS) liefern Verdachtsmomente
- Ziel: Intrusion-aware design

Komponenten eines IDS

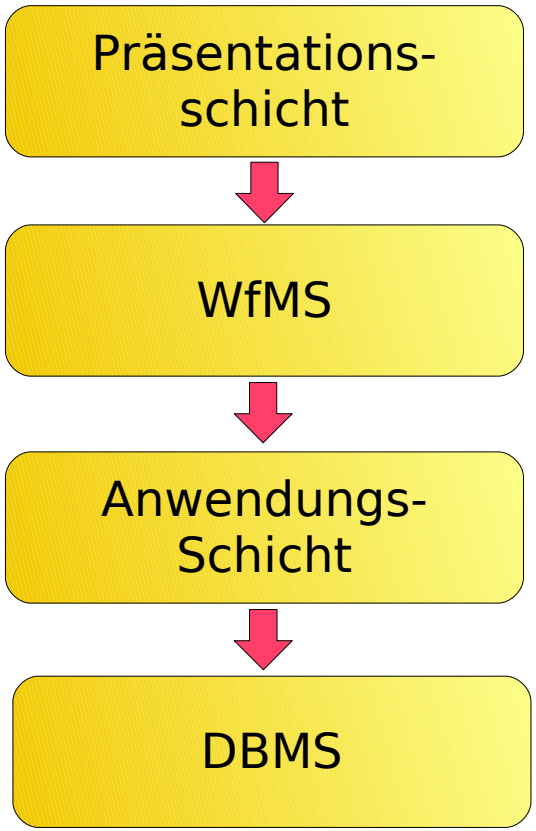


- Sensoren
  - Lieferanten der Auditdaten
  - Daten aus allen Systemebenen (BS, Netzwerk, Anwendung)
- Auswertungskomponente (Detektor)
  - statistische Anomalien-Erkennung
  - regelbasierte Erkennung
- Handlungskomponente
  - leitet die Reaktion auf die Angriffe ein
  - häufig nur geringe Fähigkeiten (z.B. Mail an Administrator, Trennung von Netzwerkverbindungen)

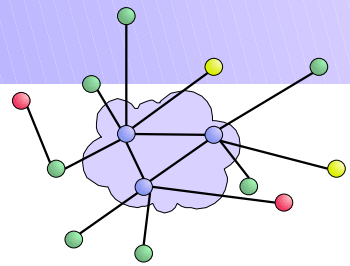
Problem: Informationsversorgung



- 1. Problem: Informationsversorgung
  - Schichten- bzw. Multi-Tier-Architektur erlaubt keine systemweite Zustandsbestimmung

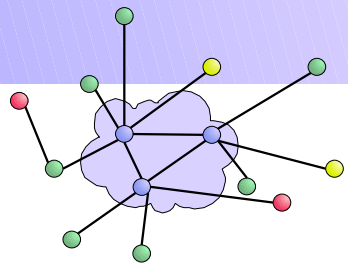


Problem: Informationsversorgung



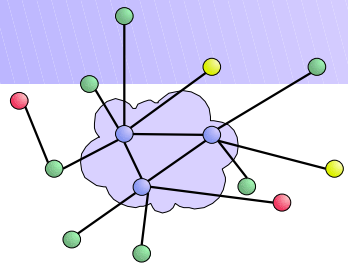
- Lösungsvorschläge
  - Aufgabe der strikten Trennung:
    - ⊕ viele Informationen zur Angriffserkennung, Handlungsspielräume
    - ⊖ hohe Kommunikationskosten, Standardplattformen
  - Erhaltung der strikten Trennung:
    - ⊕ optimierte Erkennung und Handlung, einfachere Anpassung
    - ⊖ wenig Informationen, geringe Handlungsspielräume, Aufwand
  - Kompromiss-Vorschlag: standardisierte Kommunikation zwischen den Schichten

Problem: Datenauswertung

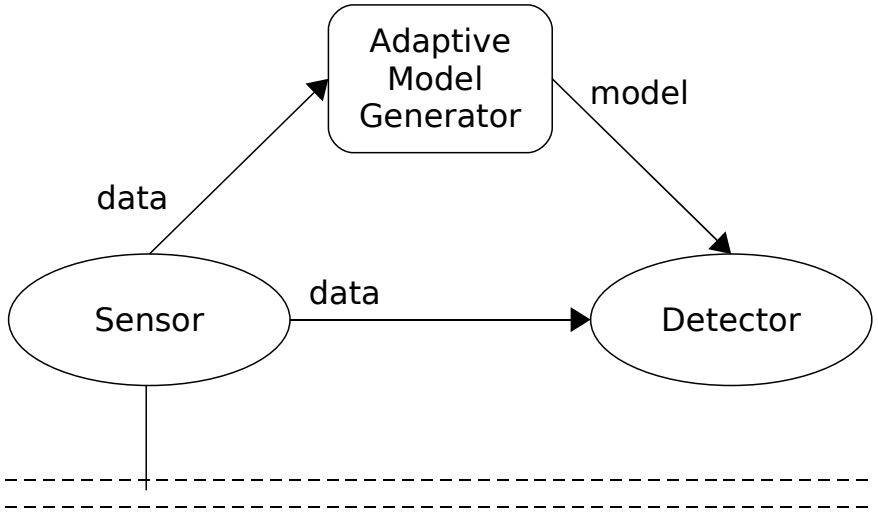


- 2. Problem: Auswertung der Auditdaten
  - Wahl des Auswertungsmodells: regelbasiert oder anomaliebasiert?
  - Welches Benutzerverhalten ist normal?
  - Was sind Anzeichen für eine Sicherheitsgefährdung?
  - Wie wird das Bewertungsmodell erstellt und wie bleibt es aktuell?
  - Typische Auditdaten (Häufigkeit und Durchsatz von Aktionen) sind in adaptiven Systemen nicht aussagekräftig!

Problem: Datenauswertung



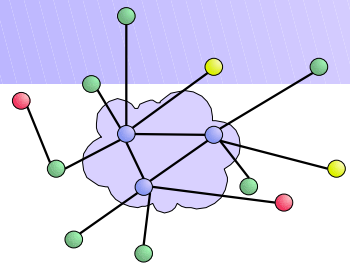
- Lösungsvorschläge:
  - Adaptiver Aufbau des Auswertungsmodells?



  - Es müssen neue Merkmale entwickelt werden!
    - Anwendungssemantik?
    - Informationen aus dem Web of Trust?

Die wichtigsten Aspekte

Fazit



- Sicherheit
  - ist die Voraussetzung für Verlässlichkeit
  - kann nur innerhalb der Systemgrenzen beeinflusst werden
  - kann nur in Form von Sicherheitsmaßnahmen spezifiziert und kontrolliert werden
- Der Entwurf sicherer Systeme kann durch Gefahrenanalysen, explizite Sicherheitsanforderungen und geeignete Werkzeuge verbessert werden
- Vertrauensmodelle bilden das Rückgrat sicherer Interaktion
- IDS benötigen ein neues Modell für die Gefahrenanalyse

Vielen Dank !

für Ihre Aufmerksamkeit

Haben Sie Fragen?