

# Betriebssysteme

## Kapitel 9 Sicherheit

## Ziele

- Sie können zwischen „security“ und „protection“ unterscheiden.
- Sie kennen die Gefährdungen von IT-Systemen.
- Sie kennen die 3-A.
- Sie kennen den Zusammenhang von Schutzmatrix und Capabilities (Fähigkeiten).

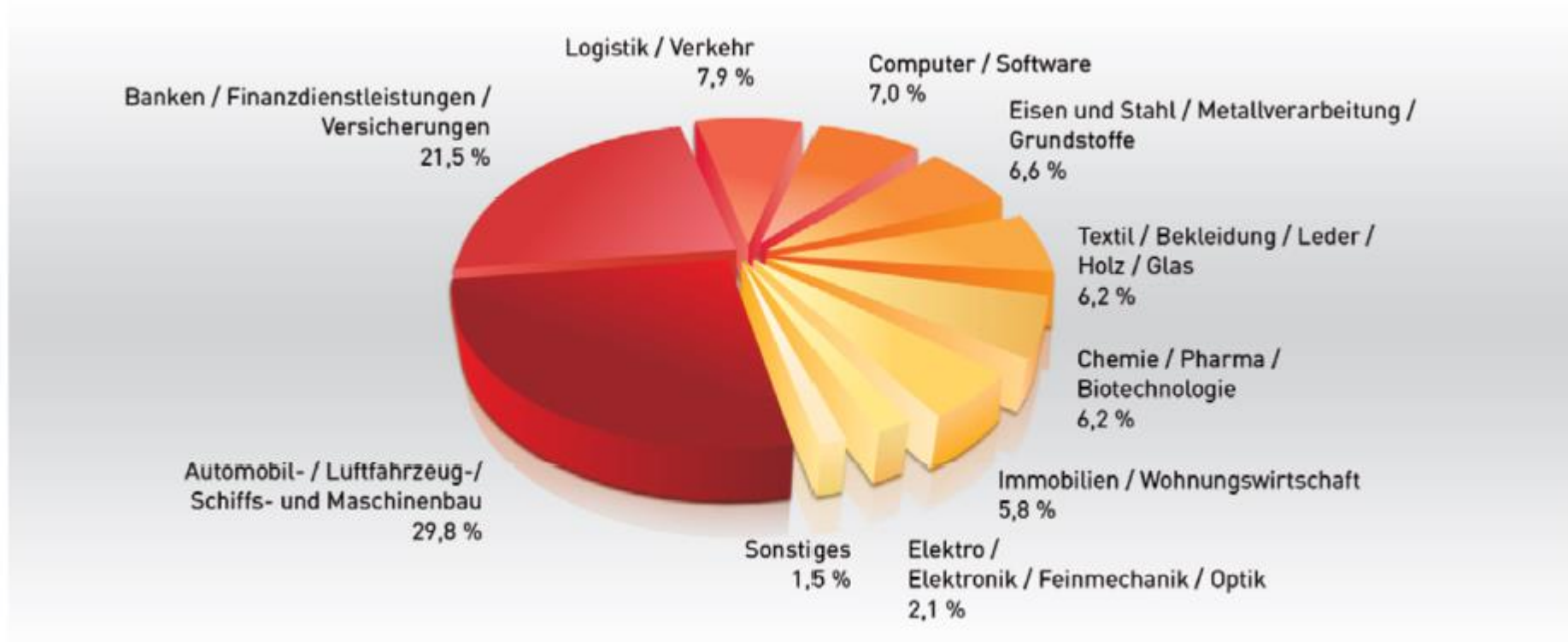
## Inhalt

- IT-Sicherheit
  - Kurzer Überblick
- Schutzmechanismen
  - Spezifisch für Betriebssysteme

- Einführung
  - Sicherheit („security“)
    - Nicht-technische Herausforderung
      - Betrachten des gesamten Prozesses notwendig!
  - Schutz („protection“)
    - Technische oder organisatorische Maßnahme
    - hier muss das Betriebssystem ansetzen
- Oft werden Sicherheit und Schutz gleichwertig verwendet
  - Eindeutige Verwendung der Begriffe hilft!



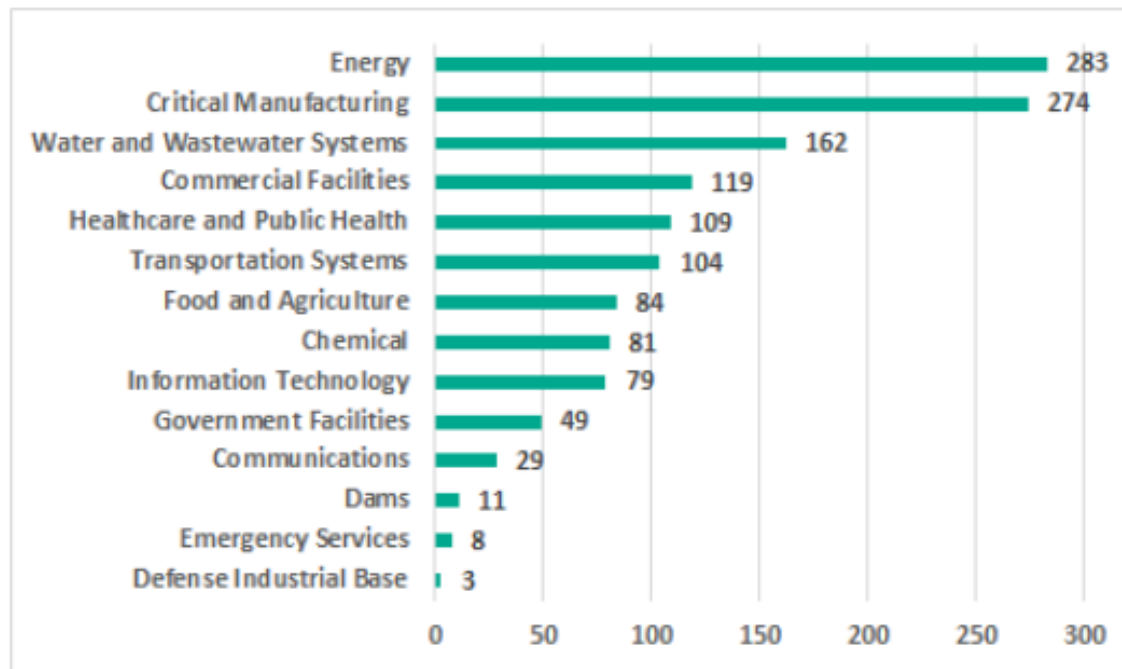
- Wer ist betroffen?



Quelle Corporate Trust 2012

# Sicherheit

- Number of vulnerable products used in different industries (according to ICS-CERT classification) vulnerabilities published in 2019



Quelle:

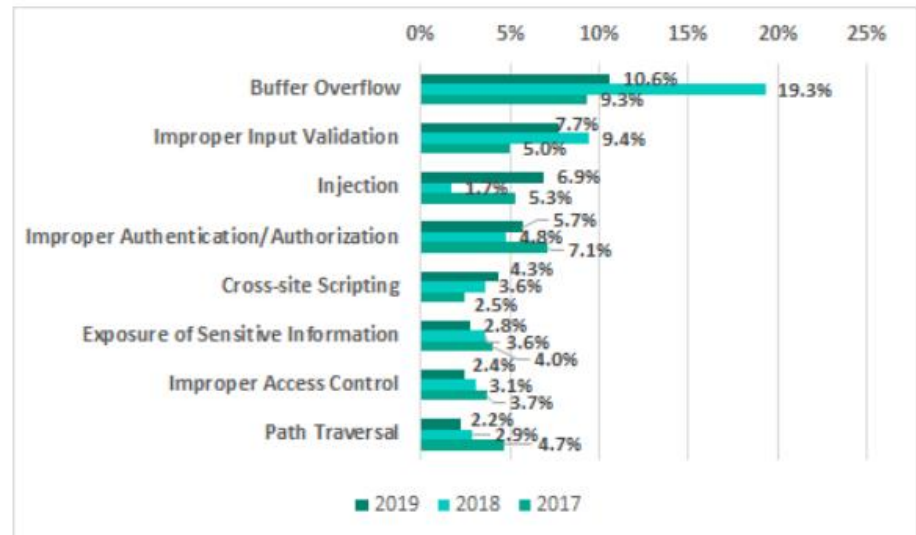
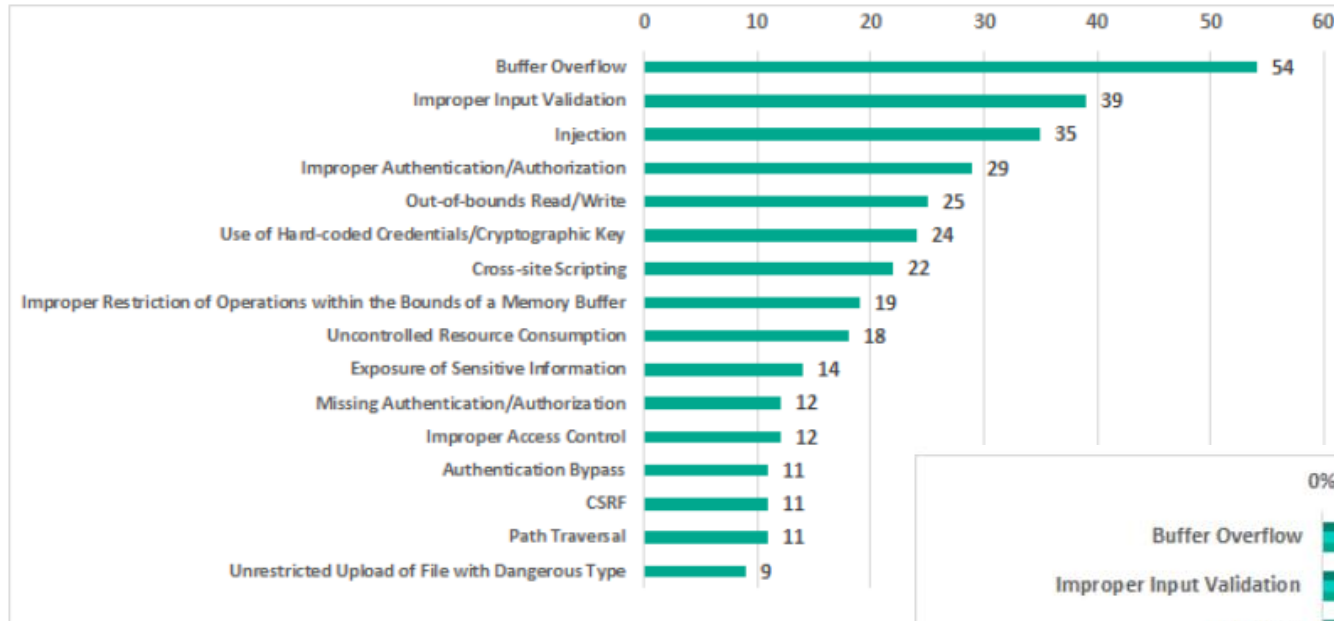
<https://ics-cert.kaspersky.com/publications/reports/2020/04/24/threat-landscape-for-industrial-automation-systems-vulnerabilities-identified-in-2019/>

<https://www.cisa.gov/known-exploited-vulnerabilities-catalog>

<https://us-cert.cisa.gov/ics>

# Sicherheit

- Most common vulnerabilities types (2019)



- **Begrifflichkeiten**

- **Sicherheit**

Sicherheit bezeichnet allgemein den Zustand, der für Individuen, Gemeinschaften sowie andere Lebewesen, Objekte und Systeme **frei von unvermeidbaren Risiken** ist oder als gefahrenfrei angesehen wird.

- **Informationssicherheit**

Als Informationssicherheit bezeichnet man Eigenschaften von informationsverarbeitenden und -lagernden (technischen oder nicht-technischen) Systemen, die die **Schutzziele Vertraulichkeit, Verfügbarkeit und Integrität** sicherstellen. Informationssicherheit dient dem **Schutz vor Gefahren bzw. Bedrohungen**, der Vermeidung von wirtschaftlichen Schäden und der Minimierung von Risiken.



- **Begrifflichkeiten**

- **Schutz**

- Schutz** ist etwas, was eine Gefährdung abhält oder einen Schaden abwehrt

- **Datenschutz**

- **Datenschutz** wird als

- Schutz vor missbräuchlicher Datenverarbeitung,
      - Schutz des Rechts auf informationelle Selbstbestimmung,
      - Schutz des Persönlichkeitsrechts bei der Datenverarbeitung und
      - Schutz der Privatsphäre verstanden

- Datenschutz wird häufig als Recht verstanden, dass jeder Mensch grundsätzlich selbst darüber entscheiden darf, **wem wann welche seiner persönlichen Daten zugänglich** sein sollen.

- Der Datenschutz soll der digitalen und vernetzten Informationsgesellschaft bestehenden Tendenz zum sogenannten gläsernen Menschen, dem Ausufern staatlicher Überwachungsmaßnahmen (Überwachungsstaat) und der Entstehung von Datenmonopolen von Privatunternehmen entgegenwirken.

- **DSGVO (Datenschutzgrundverordnung)**

- Verordnung (EU) 2016/679 des Europäischen Parlaments und des Rates vom 27. April 2016 zum Schutz natürlicher Personen bei der Verarbeitung personenbezogener Daten, zum freien Datenverkehr und zur Aufhebung der Richtlinie 95/46/EG

- **Sicherheits-Ziele und Bedrohung des Betriebssystems**
  1. **Vertraulichkeit**
    - geheime Daten sollen auch geheim bleiben
    - Bedrohung: Enthüllung der Daten
  2. **Integrität**
    - Unautorisierte Benutzer oder auch Systemdienste sollen nicht in der Lage sein, Daten ohne die Erlaubnis des Besitzers zu modifizieren.
    - Bedrohung: Manipulation der Daten (Datenintegrität)
  3. **Verfügbarkeit (Systemverfügbarkeit)**
    - Niemand darf das System so stören, dass es dadurch nur noch eingeschränkt oder sogar vollkommen unbenutzbar wird.
    - Bedrohung:
      - Dienstverweigerung (denial of service=DOS, DDoS=Distributed Denial of Service)
      - Systemübernahme durch Viren

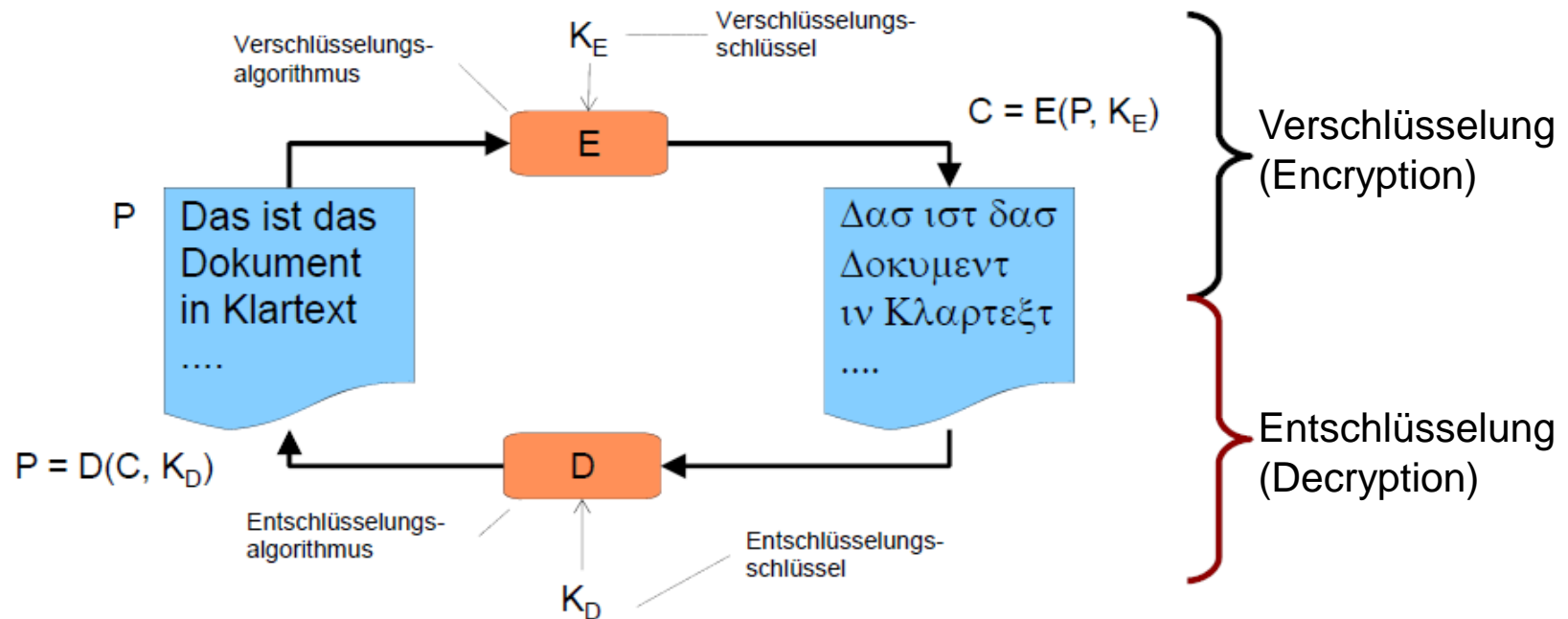
- Gefahren für Systeme
  - durch unbeabsichtigten Datenverlust
    - Katastrophen
    - Hardware- und Softwarefehler
    - Menschliches Versagen
  - durch Angreifer
    - Passive Angreifer
      - interessiert an Daten
    - Aktive Angreifer
      - Manipulation von Systemen und Daten



**Es gibt wahrscheinlich mehr unbeabsichtigte Schäden als Schäden durch Angreifer**

# Sicherheit

- Einführung in die Kryptographie



1. Security by **Obscurity**: Halte den Algorithmus geheim!
  - Auf Dauer nicht geeignet für Sicherheitsmaßnahmen
2. **‘Geheimhaltung in den Schlüsseln verstecken’** (Algorithmen öffentlich) (Kerckhoffs’ Maxim, 1883)

# Sicherheit

- Einführung in die Kryptographie
  - **Symmetrische Kryptographie**
    - Gleicher Schlüssel zur Ver- und Entschlüsselung
    - Sowohl Sender und Empfänger müssen im Besitz des geheimen Schlüssels sein

➡ Übertragung muss über sicheren Kanal geschehen!

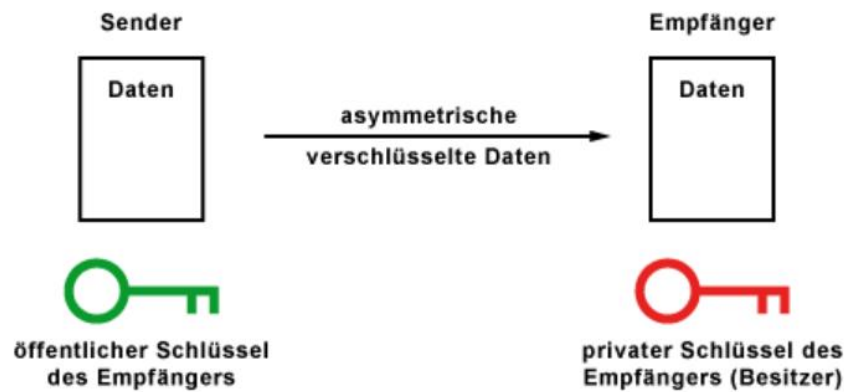


Beispiele: AES (advanced encryption standard), DES (data encryption standard), 3DES

Quelle: <https://www.elektronik-kompodium.de/sites/net/1910101.htm>

- Einführung in die Kryptographie
  - **Asymmetrische Kryptographie**
    - Schlüsselpaar aus geheimem (privatem) und öffentlichem Schlüssel
      - Verschlüsselung mit öffentlichem Schlüssel
      - Entschlüsselung mit geheimem Schlüssel
        - › Public-Key-Verfahren (Diffie und Hellman, 1976)
        - › Sender verwendet den öffentlichen Schlüssel, der Empfänger kann an ihn adressierte verschlüsselte Nachrichten entschlüsseln

➡ Wichtig ist die sichere Aufbewahrung des privaten Schlüssels!!




Beispiel: RSA (Rivest, Shamir, Adleman)

Quelle: <https://www.elektronik-kompodium.de/sites/net/1910101.htm>

- Sicherheitsmechanismen
  - „Triple A“ - AAA (Authentifizierung, Autorisierung, Accounting)
    - **Authentifizierung**
      - Wer macht was?
      - Feststellung der Identität durch Prüfung durch Passwörter, durch Besitz (z.B. Speicherchipkarten, el. Tokens, ...), durch biometrische Erkennung
    - **Autorisierung**
      - Wer darf was im System tun?
      - Vergabe von Zugriffsrechten an Benutzer
    - **Accounting**
      - Wer hat was im System gemacht?
      - Protokollierung von Aktivitäten

- Schutzmechanismen

- Computersystem enthält viele Ressourcen (Objekte), die geschützt werden sollen (z.B. CPUs, Speicher, HDDs,..., Prozesse, Dateien, DBs,...)
- Jedes Objekt hat einen eindeutigen Namen, über den es angesprochen wird und eine endliche Menge an Operationen, die von Prozessen auf diesem Objekt ausgeführt werden können.

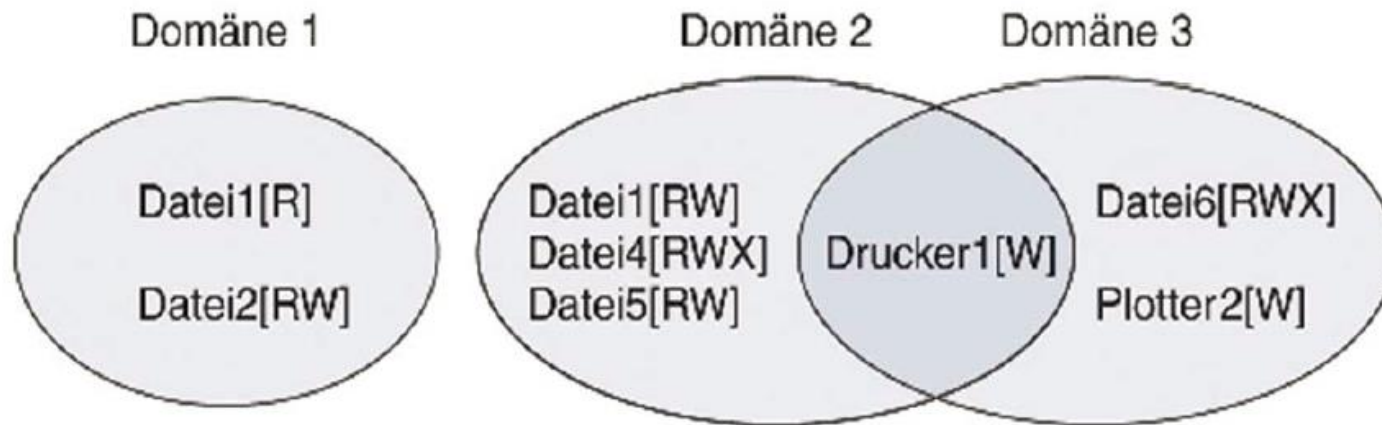
 Verfahren notwendig, das den Zugriff von Prozessen auf Objekte verhindert (z.B. Prozess A darf Datei X lesen, aber nicht schreiben)

- „Triple A“ - AAA (Authentifizierung, Autorisierung, Accounting)



- Schutzmechanismen
  - Schutz-Domäne (domain)
    - Was ist zu schützen?
      - Hardware: CPUs, Speichersegmente, Platten, Drucker,...
      - Software: Dateien, Prozesse, Datenbanken, Semaphoren,...
    - Schützenswerte Objekte
      - Eindeutige Namen
      - Endliche Menge von Operationen, z.B. Für Dateien: read und write; für Semaphoren: up und down
    - Recht
      - Erlaubnis eine Operation durchzuführen
  - **Domäne** ist Menge von (Objekt, Rechte)-Paaren
    - Beispiel: Benutzer, Benutzergruppe, Prozess, ...

- Schutzmechanismen
  - Schutz-Domäne (domain)
    - Jeder Prozess läuft zu jedem Zeitpunkt in einer bestimmten Schutzdomäne



Domäne = Kreise

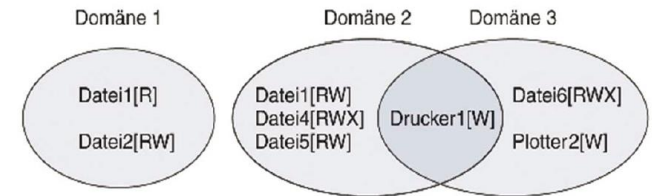
Objekte = Datei1, Datei2, Drucker1, ...

Rechte = R, W, X

- Schutzmechanismen

- Schutz-Matrix

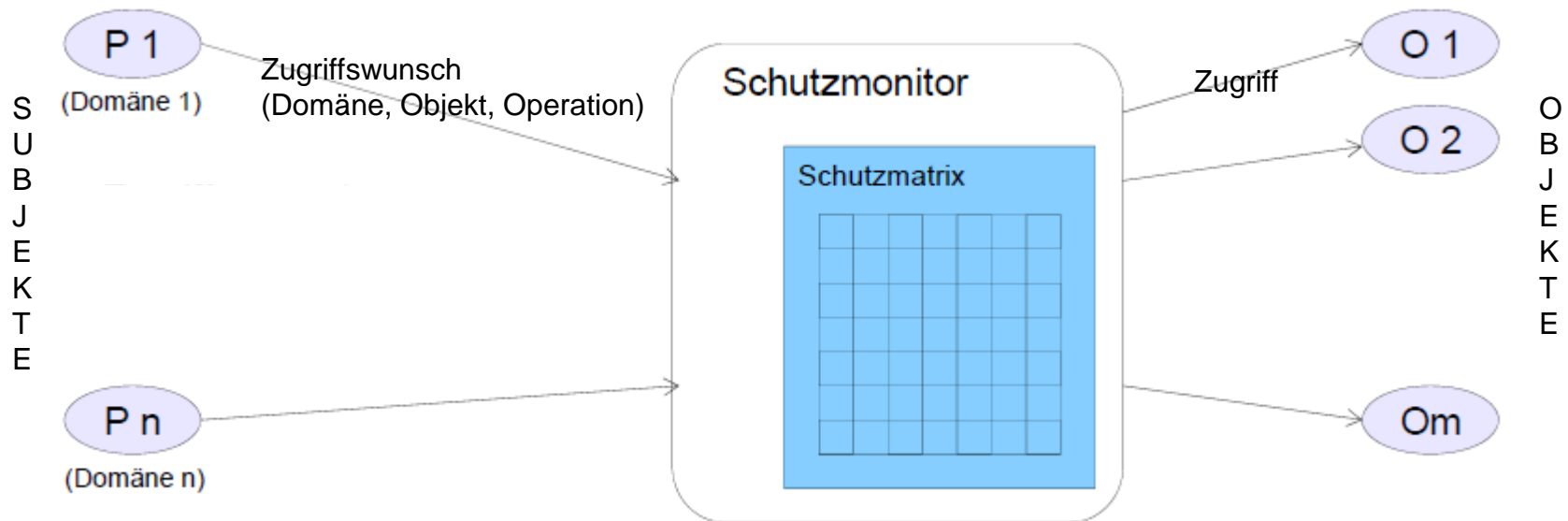
- Anzeige welches Objekt zu welcher Domäne gehört
    - Matrix und Domänennummer entscheidet, ob ein bestimmter Zugriff auf ein Objekt von einer bestimmten Domäne erlaubt ist.



		Objekt							
		Datei1	Datei2	Datei3	Datei4	Datei5	Datei6	Drucker1	Plotter2
Domäne	1	Read	Read Write						
	2			Read	Read Write Execute	Read Write		Write	
	3						Read Write Execute	Write	Write

- Schutzmechanismen

- Realisierung der Zugriffskontrolle über Schutzmonitor
  - Subjekte dürfen Domäne nicht selbständig wechseln können
  - Zugriff auf Objekte darf nur über Schutzmonitor möglich sein
  - Schutzmonitor muss vertrauenswürdig sein
  - Schutzmonitor muss privilegiert sein, um Zugriffe durchzuführen



# Aufgabe/Frage

Wir haben ein sehr einfaches System

- Domänen = (Objekt, Rechte)-Paare
- Schutzmatrix
- Zugriffsmonitor

kennengelernt.

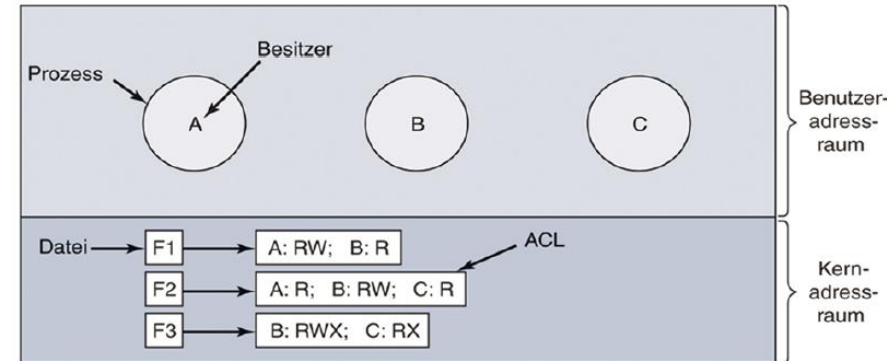
- Überlegen Sie, was der Nachteil dieses Systems in großen Systemen mit vielen Benutzern ist.

Bedingung:  
→ ad hoc

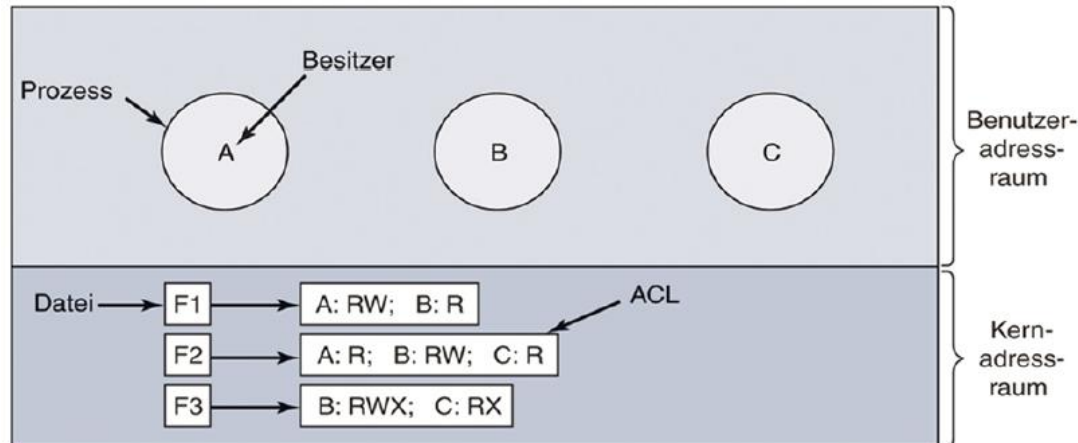


ad hoc

- Zugriffskontrollliste (ACL)
  - Spalte der Schutzmatrix
  - Gibt für ein Objekt an, welche Subjekte welche Rechte an dem Objekt haben
  - Wird zusammen mit dem betroffenen Objekt gespeichert
    - z.B. bei Datei im zugehörigen I-Node
  - Listenelemente: Paare (Subjekt, Rechte)
    - Subjekt: Benutzer und/oder Benutzergruppe
    - für Subjekt oft auch Platzhalter (Wildcard) erlaubt
      - erster passender Eintrag wird verwendet



- Zugriffskontrollliste – Access Control List (ACL)



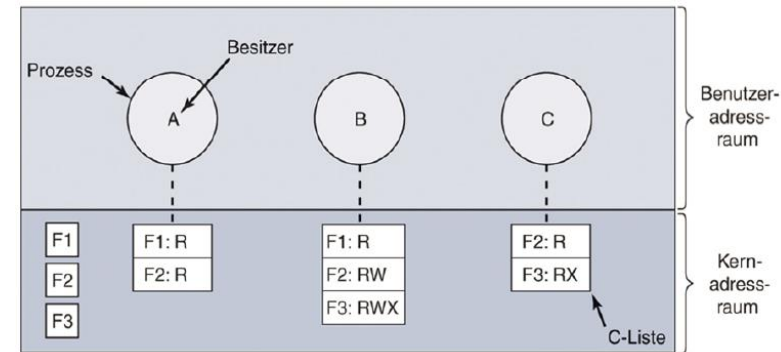
- 3 Prozesse, jeder gehört zu einer unterschiedlichen Domäne (A, B, C)
  - 3 Dateien (F1, F2, F3) können besessen werden
  - Vereinfachung; jede Domäne gehört genau zu einem Benutzer (oft Subjekte genannt)
  - Jeder Datei ist eine ACL zugeordnet
    - F1 besitzt in ACL zwei Einträge, getrennt mit Semikolon
      - Jeder zu Benutzer A gehörende Prozess darf die Datei lesen und schreiben
      - Jeder zu Benutzer B gehörende Prozess darf die Datei lesen
      - Alle weiteren Zugriffe von diesen Benutzern sowie sämtliche Zugriffe von anderen Benutzern sind verboten
- **Achtung:** Rechte werden nach Benutzern und nicht nach Prozessen vergeben.

- Zugriffskontrollliste

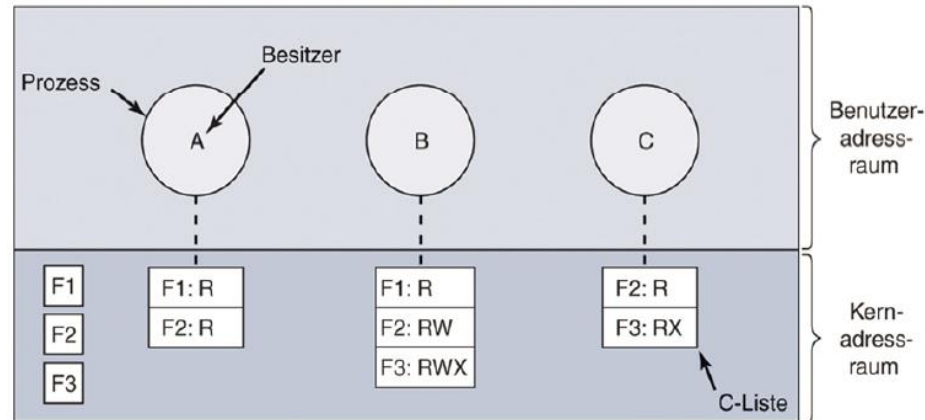
- Es gibt auch universelle Rechte (Bezug auf alle Objekte, egal welchen Typs)
  - z.B. Destroy object, copy object
- Es gibt auch objektspezifische Rechte
  - z.B. Append message für mailbox-Object, sort alphabetically für Verzeichnis-Object
- Auch Benutzergruppen möglich
  - Gruppen besitzen Namen
  - Können in ACLs enthalten sein
  - Jeder Prozess besitzt eine Benutzer-ID (UID) und eine Gruppen-ID (GID)
    - ACL-Eintrag: UID1,GID1: RECHTE1; UID2,GID2:RECHTE2;...
    - Ist bei Zugriff auf ein Object UID und GID in der ACL vorhanden, dann RECHTE verfügbar; wenn nicht Zugriff verweigert.
- Mit der Nutzung von Gruppen wird das **Konzept der Rolle** eingeführt.



- Zugriffskontrollliste (capability)
  - Übersetzung capability: Befähigung, Fertigkeit, Leistungsfähigkeit,
  - Zeile der Schutzmatrix
  - Wird vom Betriebssystemkern an Subjekte (Prozesse) übergeben, berechtigt zur Ausführung von Operationen auf Objekten
  - Jeder Domäne wird eine Liste zugeordnet, die die Objekte und Operationen enthält, auf die die Domäne zugreifen kann
    - Capability-Liste (C-Liste)
  - Capability muß vor Manipulation geschützt werden!
    - C-Liste Speicherung im Betriebssystemkern, Prozeß erhält nur Verweis (Handle)
    - kryptographischer Schutz (analog zu digitaler Signatur)
      - geeignet für verteilte Systeme: Capability kann als Nachricht weitergegeben werden



- Capabilities



- Eine Capability besteht aus einem Datei (Objekt-)Identifikator und einer Bitmap für die verschiedenen Rechte. z.B. Paare (Objekte, Operationen)
- Jede capability gewährt ihrem Besitzer gewisse Rechte auf bestimmte Objekte
- Beispiel:
  - 3 Prozesse und ihre capability-Liste (A, B, C)
  - Der zu Benutzer A gehörige Prozess kann die Dateien F1 und F2 lesen
- Capability-Listen sind selbst auch Objekte; andere capability-Listen können darauf zeigen

- Vergleich Capabilities und ACLs
  - Capabilities sind effizient
    - keine Überprüfung, wenn ein Prozess Zugriff auf eine Datei fordert
    - Im Gegensatz zu evtl. langer Suche in ACLs
    - Wenn keine Gruppen unterstützt werden, müssen alle Benutzer in der ACL aufgelistet werden, um jedermann Lesezugriff auf eine Datei zu geben.
  - Capabilities können Prozesse effizient kapseln; ACL nicht
  - ACL erlauben den selektiven Entzug von Rechten
    - z.B.: Löschen eines Objekts ohne Löschung der Capability kann Probleme machen; auch umgekehrt
  - Capabilities und ACLs ergänzen sich
  - **Realisierung durch Mischform**
    - ACL für Rechteverwaltung
    - Prüfung nur beim Öffnen, danach wird ein Handling wie eine Capability verwendet.

- Implementierung der Zugriffssteuerung
  - POLA = Principle of Least Authority
    - Sicherheit funktioniert am besten, wenn jede Domäne
      - die minimalen Privilegien und
      - die minimale Anzahl an Objekten hat, die notwendig sind um ihre Aufgabe zu erfüllen.

- Implementierung
  - Massnahmen gegen Angriffe
    - **Insider Angriffe**
      - Der Code wird bei Erstellung des Systems eingebaut
      - Fall-Türen, Login-Spoofing,...
      - Gegenmassnahme: Code-Review
    - **Ausnutzen von Programmierfehlern**
      - Pufferüberläufe, Code-Injektion
      - Gegenmassnahme: Programmierempfehlungen
    - **Injektion von Malware**
      - Der Code wird später ins laufende System gebracht
      - Viren, Würmer, Trojaner,...
      - Gegenmassnahmen: Firewalls, Anti-Virensoftware,...

- Beispiel: Code-Injection

- **Einschleusen und Ausführung von Code** auf den Zielsystem oder in ein Programm.
- Ziel: Manipulation von Einträgen in der Datenbank, von Skripten und Dateien.
- Vorkommen: **Überall dort, wo Anwender selbst Einträge vornehmen und Informationen übertragen können.** z.B.: Beim Cross Site Scripting (XSS) in Bereichen dynamischer Webseiten (aktive Angriffsform).

Beispiel:

Browser stellt manipulierte Webseite vollständig dar, er erkennt den tatsächlichen Zweck des enthaltenen Codes nicht.

Schadhafter Code ist Teil der Seite, Sicherheitsvorkehrungen wie das Einsetzen von Verschlüsselungstechniken helfen nicht.

- **Mögliche Erscheinungsformen:** SQL Injection, Webscript Injection, XML Injection, JSON Injection, OS Command Injection und YPATH Injection.
- Typische Angriffsziele: Anwendungen, die die übermittelten Daten kaum auf Risiken prüfen, z.B. Foren, Gästebücher, private Nachrichten,...

- Zusammenfassung
  - Ziele und Bedrohungen
  - Kryptographie
  - Sicherheitsmechanismen
    - AAA
  - Schutzmechanismen und Domänen
    - Sicherheitsmatrix
    - Zugriffslisten (ACL)
      - verbunden mit einem Objekt
    - Capability-Listen
      - verbunden mit einer Domäne

# Aufgabe/Frage

- Wir haben über verschiedene Angriffsziele und -klassen gesprochen. Beschreiben Sie kurz jede der Angriffsklassen und überlegen Sie sich, welche der Schutzziele durch einen solchen Angriff potentiell gefährdet sind.

Angriffsklasse	Beschreibung	Inte- grität	Vertrau- lichkeit	Verfüg- barkeit	Verbind- lichkeit	Authen- zität	Privat- heit
Sniffen							
Spoofen							
DoS							
XSS							
SQL-Injection							
Viren							
Würmer							
Trojaner							
Phishing							
Spamming							
Social Engineering							

Bedingung:

→ 10 min

10 min



Angriffsklasse	Beschreibung	Inte- grität	Vertrau- lichkeit	Verfüg- barkeit	Verbind- lichkeit	Authen- zität	Privat- heit
Sniffen							
Spoofen							
DoS							
XSS							
SQL-Injection							
Viren							
Würmer							
Trojaner							
Phishing							
Spamming							
Social Engineering							