

# IT-SICHERHEIT

## 0. EINFÜHRUNG

Wintersemester 2015/16

Prof. Dr. Michael Meier  
Informatik IV  
Arbeitsgruppe IT-Sicherheit



Universität Bonn  
E-Mail: [mm@cs.uni-bonn.de](mailto:mm@cs.uni-bonn.de)

# ZUR PERSON: PROF. DR. MICHAEL MEIER

- 1993 – 1998      **Studium** der Informatik an der  
TU Cottbus
- 1999 – 2005      **Wissenschaftlicher Mitarbeiter** an der  
TU Cottbus, Lehrstuhl Rechnernetze
- 2006                **Promotion** zum Dr. rer. nat.  
(IT-Sicherheit / Intrusion Detection)
- 2006 – 2010      **Wissenschaftlicher Mitarbeiter**  
TU Dortmund, AG Informationssysteme und Sicherheit
- 2010 – 2011      **Professurvertretung**  
Uni Bonn
- Seit 2012          **Professur für IT-Sicherheit**  
Uni Bonn  
Abteilungsleiter Cyber Security  
bei Fraunhofer FKIE



# WER SIND SIE?

3

# ALLGEMEINE HINWEISE I

- Dies ist die erste Auflage dieser Vorlesung
  - Keine perfekt polierten Materialen erwarten
  - Bitte geben Sie uns Rückmeldung, was in künftigen Auflagen verbessert/geändert werden sollte
- Die Vorlesung führt in das Gebiet IT-Sicherheit ein
- Voraussetzungen
  - Grundlegende Kenntnisse in den Bereichen Betriebssysteme und Computernetzwerke
- Termine
  - Vorlesungen: wöchentlich donnerstags 16.30 – 18.00 HS1
  - Übungen: wöchentlichen donnerstags 14.30 – 16.00 HS1
  - Keine Vorlesung und keine Übung am 29.10.
  - Keine Übung am 5.11.

4

# INHALT

## ■ Geplante Inhalte

- Informationstechnik: Computer und Netze
- Grundlegendes zu IT-Sicherheit
- Authentifikation
- Zugriffskontrolle
- Angewandte Kryptographie
- Sicherheitsmanagement

5

# ZUR ÜBUNG

- Übungsleiter: Michael Meier und Mitarbeiter der AG IT-Sicherheit
- Wöchentlich donnerstags wird ein Übungsblatt herausgegeben,
  - das zuhause bearbeitet werden soll und
  - in der darauf folgenden Woche in der Übung besprochen wird.
    - Sie stellen Ihre Lösungen vor und diskutieren diese mit Ihren Kommilitonen
  - Keine Abgaben
- Übungsinhalte
  - Festigung und Ergänzung des Vorlesungsstoffes
  - Kennenlernen und Verwendung relevanter Werkzeuge
  - Studium und Diskussion ausgewählter Literatur

# ALLGEMEINE HINWEISE II

- Website:
  - <https://net.cs.uni-bonn.de/wg/itsec/teaching/wt-201516/ba-inf-138-it-sicherheit/>
- Anmeldung
  - Melden Sie sich bis 25.10. in TVS für die Veranstaltung an
  - Mailing-Liste (wichtig!):
    - Mit Ihren TVS-Anmeldedaten (Email-Adresse) werden Sie in die Mailing-Liste zur Vorlesung eingetragen
    - Über die Mailing-Liste erhalten Sie wichtige Informationen zur Vorlesung und Übung und können selbst mit den Teilnehmer der Vorlesung diskutieren
- Materialen (Folien, Übungsblätter)
  - Werden auf der Webseite verfügbar gemacht

Login: itsi

Password: wird/wurde in der Vorlesung  
am 22.10. bekannt gegeben

# ALLGEMEINE HINWEISE III

- Studienleistungen
  -
- Prüfung
  - Schriftliche Prüfungsklausur
  - Zu den Inhalten der Vorlesung und Übung

8

# FRAGEN?

9

# IT-SICHERHEIT

## 1. INFORMATIONSTECHNIK – COMPUTER UND NETZE

Wintersemester 2015/16

Prof. Dr. Michael Meier  
Informatik IV  
Arbeitsgruppe IT-Sicherheit



Universität Bonn  
E-Mail: mm@cs.uni-bonn.de

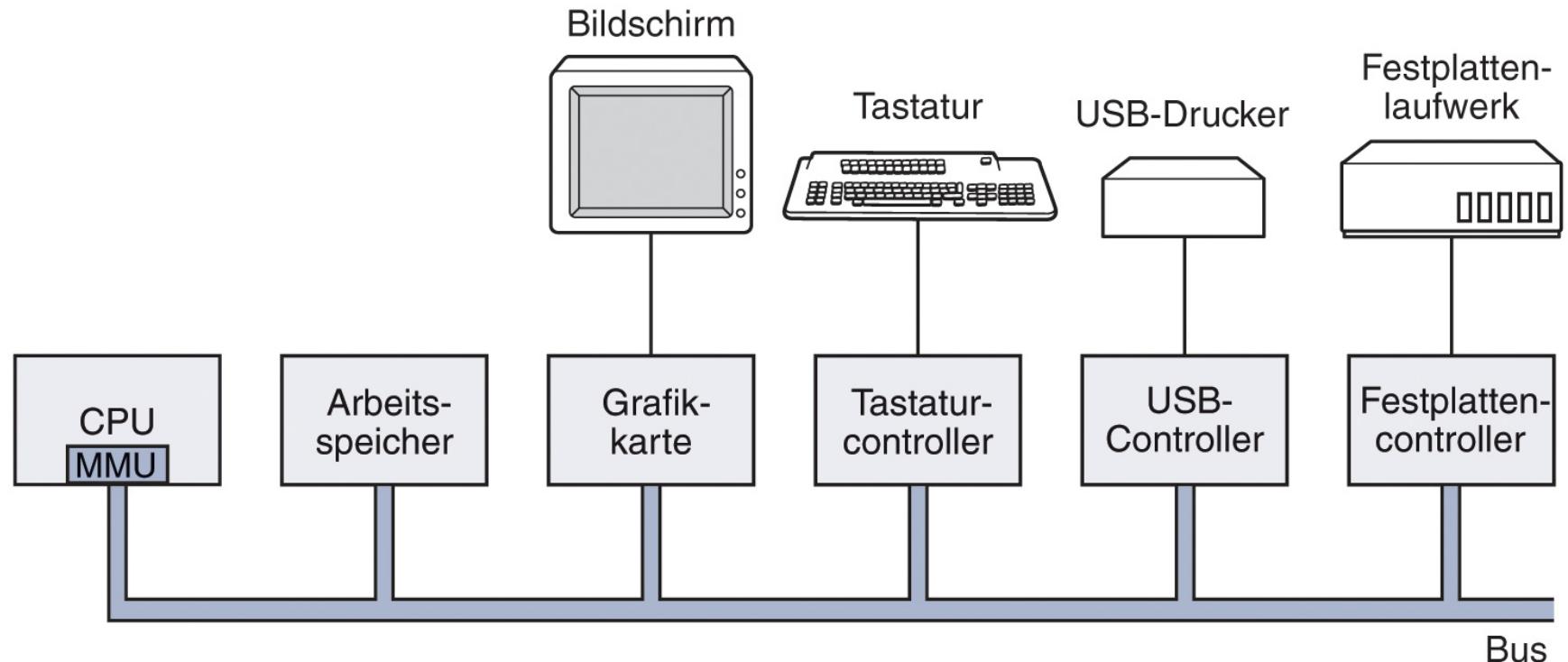
# ÜBERBLICK

- Wiederholung und Festigung grundlegender Kenntnisse
- Vereinfachter Überblick
  - Aufbau und Abläufe von Computersystemen
  - Aufbau und Funktionsweise von Computernetzwerken

2

# ÜBERBLICK ÜBER DIE COMPUTER-HARDWARE

## ■ Komponenten eines einfachen PC



# CPU (x86 FAMILIE)

- „Hirn“ des Rechners: holt Instruktion aus Speicher und führt sie aus
- CPU-Betriebsmodi (operating modes) zwischen denen gewechselt werden kann
  - Real Mode (eigentlich Real Address Mode)
    - x86 CPUs starten nach Reset im Real Mode
  - Protected Mode (eigentlich Protected Virtual Address Mode)
    - Ab 80286/386er verfügbar
  - Es gibt weitere Modi
    - zum Beispiel System Management Mode (SMM)

# CPU (x86 FAMILIE) - REAL MODE

## ■ Real Mode

- x86 CPUs starten nach Reset im Real Mode
- Kein Zugriffsschutz:
  - jedes laufende Programm hat vollen Zugriff auf gesamten Speicher und gesamte Hardware
- ~1 MB Speicher adressierbar
- Standard-Modus für Betriebssysteme wie MS-DOS

# CPU (x86 FAMILIE) - PROTECTED MODE

- Die meisten modernen Betriebssysteme arbeiten im Protected Mode (z.B. Windows, Linux, Mac OS X)
- Unterscheidet 4 Privilegienebenen / Ringe
  - Ring 0 (hoch privilegiert): alle Instruktionen verfügbar und alle Speicheradressen zugreifbar
  - Ring 3 (wenig privilegiert): I/O Instruktionen verboten
  - Ermöglicht die Unterscheidung von Kernel-Modus (Ring 0) und Benutzer-Modus (Ring 3) in Betriebssystemen
- Unterstützt Paging zur Speicherverwaltung
  - Voneinander getrennte virtuelle Adressräume für Prozesse
  - Trennung zwischen Kernel- und Benutzeradressraum
  - Speicherschutz
  - 16 MB (16bit) bzw. 4 GB (32bit) adressierbar

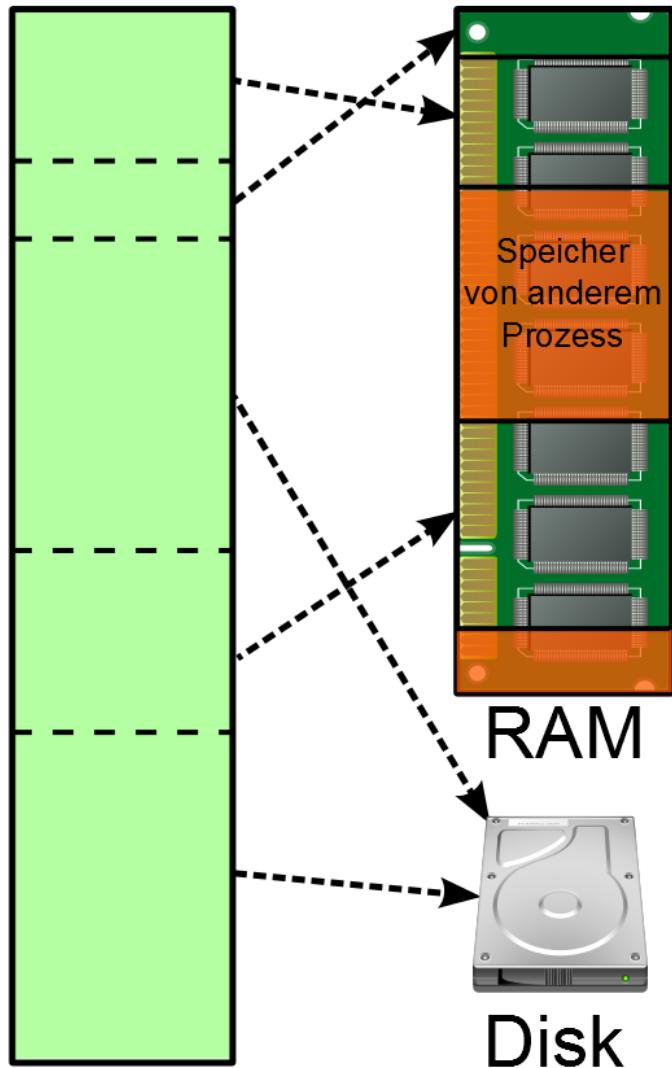
- Notwendigkeit von Speicherschutz
  - Betriebssystemkern ist privilegiert
  - Benutzer-Anwendungsprogramme sind nicht-vertrauenswürdig
    - Sicherheit: maliziöse Programme lesen/schreiben Daten
    - Zuverlässigkeit: fehlerhafte Programme können gesamten Rechner crashen
- ⇒ Betriebssystemkern muss vor Benutzer-Anwendungsprogrammen geschützt werden
- ⇒ Benutzer-Anwendungen sollen voreinander geschützt werden

# CPU – PROTECTED MODE

- Jedem Prozess kann eigener virtueller Speicherbereich zugordnet werden
- Zugriff auf Speicher anderer Ressourcen und Prozesse resultiert in Fehler/Schutzverletzung
- Speicherauslagerung möglich

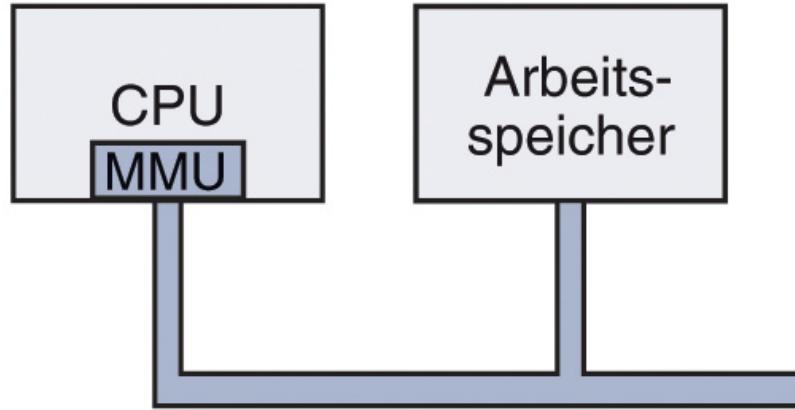
Virtueller Speicher  
(pro Prozess)

Physischer Speicher



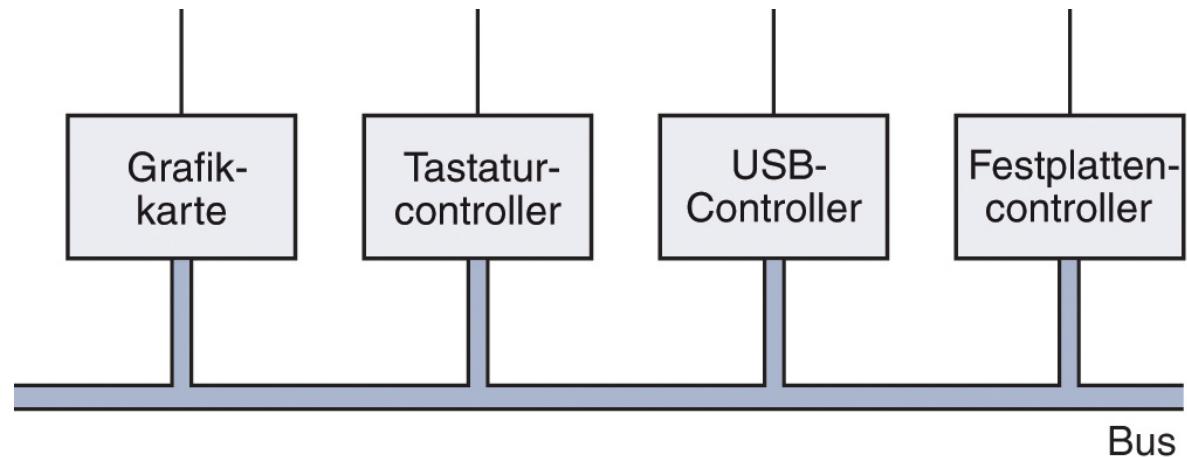
# MMU – MEMORY MANAGEMENT UNIT

- Virtuelle Adressierung
  - Virtueller Speicher eines Prozesses kann größer sein als physischer Speicher
- MMU übersetzt on-the-fly virtuelle Prozessadressen in physische RAM Adressen



# HARDWARE-CONTROLLER

- Hardware-Controller haben eigenen Speicher
- Eine Controllerkarte enthält zwei unterschiedliche Arten Programmcode
  - Firmware (Festware) zum Ansteuern der Hardware-eigenen Bausteine
  - Option-ROM mit Programmcode, der im Wesentlichen zur Initialisierung durch die CPU ausgeführt wird



# FESTSPEICHER

- Typischerweise organisiert in aufeinanderfolgenden Sektoren (traditionell 512 Byte)
- Ggf. unterteilt in mehrere Partitionen
  - Partition: zusammenhängende Sektoren
- Bootbares Speichermedium enthält Master Boot Record (MBR)
  - Steht im ersten (nullten) Sektor des Speichermediums
  - 512 Byte groß mit Magic Bytes 0x55AA am Ende
  - Enthält
    - Partitionstabelle und
    - Master Boot Code, der Partitionstabelle liest und von einer eingetragenen Partition einen Bootloader startet

# FESTSPEICHER - MBR

- MBR enthält Code und Partitionstionstabellle
- Anhand MBR-Signatur „0x55AA“ kann ein MBR erkannt werden

Disk mit N Sektoren a 512 Bytes

The diagram illustrates the layout of a disk. At the top, a large rectangular box represents the disk, divided into five horizontal sections labeled "Sektor 0 MBR", "Sektor 1", "Sektor 2", "...", and "Sektor N-1". A diagonal line extends from the bottom right corner of the "Sektor 0 MBR" section down to the bottom left corner of the "Master Boot Record (512 Bytes)" box below. This indicates that the MBR is located at the start of the first sector.

Master Boot Record (512 Bytes)

Master Boot Code  
(440 Bytes)

Disk-  
Signatur  
(4 Bytes)

Nullen  
(2  
Bytes)

Partitionstabelle  
(4 Einträge  
a 16 Byte)

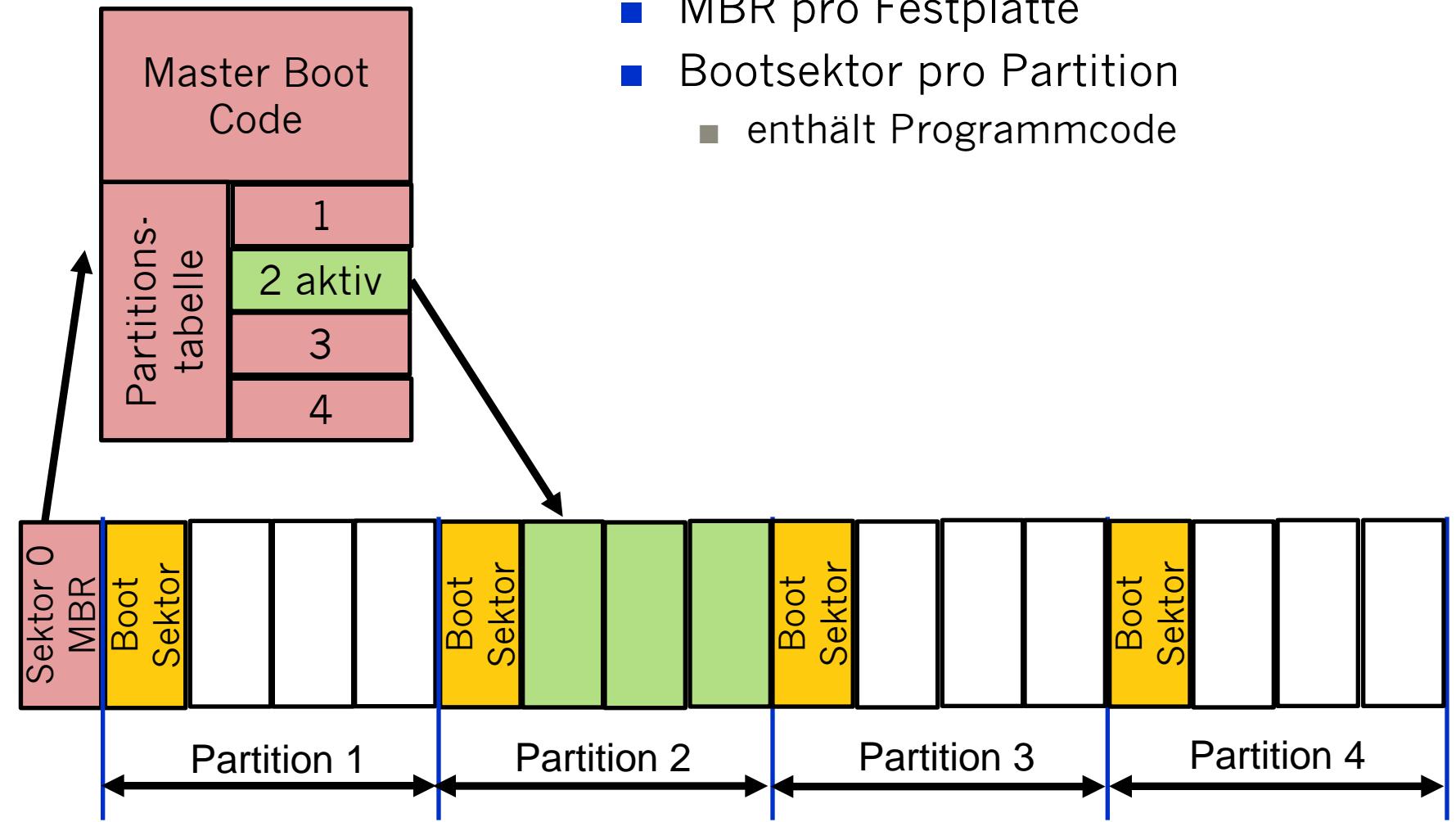
MBR Signatur  
(2 Bytes)  
„0x55AA“

12

# FESTSPEICHER – MBR UND BOOTSEKTOR

13

- MBR pro Festplatte
- Bootsektor pro Partition
  - enthält Programmcode



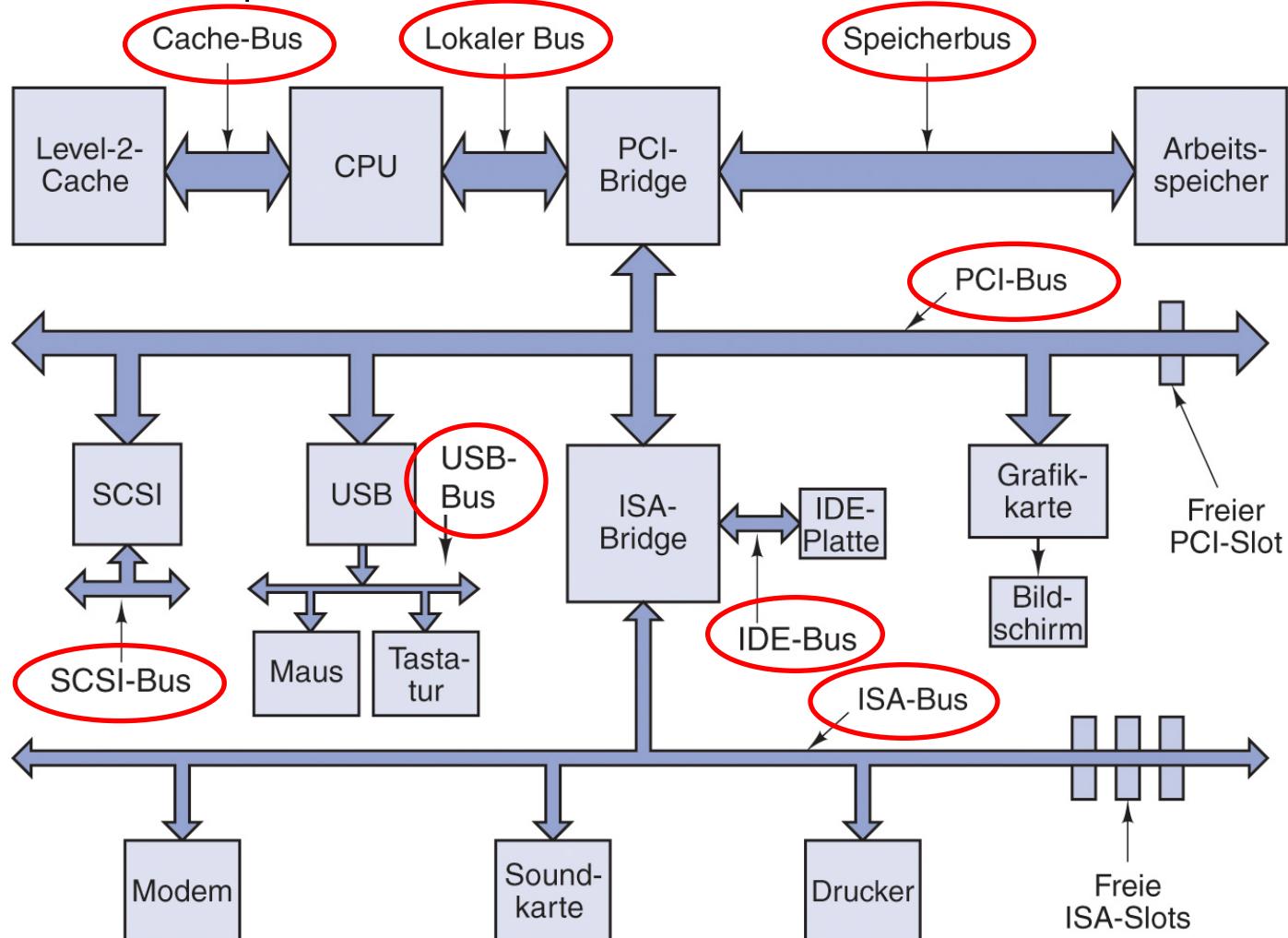
- Um zuhause Besuchern die Tür zu öffnen können Sie
  - vor der Tür sitzen und warten bis der Besucher kommt (Busy Waiting)
  - regelmäßig an der Tür schauen, ob ein Besucher da ist (Polling)
  - eine Klingel installieren und bei Ertönen des Klingelzeichens zur Tür gehen (Interrupt)

# INTERAKTIONSSTEUERUNG DURCH INTERRUPTS

- Interaktion und Wechsel zwischen Geräten und Prozessen durch Interrupts
  - Ein/Ausgabegeräte signalisieren der CPU Anforderungen/Ergebnisse
    - Die Netzwerkkarte hat neue Daten empfangen
  - Zugriffsfehler werden der CPU signalisiert
    - Illegaler Speicherzugriff durch einen Prozess
  - Timer-Ereignisse werden der CPU signalisiert
    - Das Betriebssystem hat regelmäßig Aufgaben zu erledigen, wozu laufende Programme unterbrochen werden.
  - Je nach Art des Interrupts wird entsprechende Interrupt-Behandlungsroutine (Instruktionssequenzen) ausgeführt

# BUSSE

- Hardware-Komponenten kommunizieren über ein Bus-System
- Aufgrund gestiegener Leistung der Komponenten verfügen moderne Computer über mehrere Busse



# ABLAUF BEIM BOOTING

- Auf dem Parent-Board (früher Mother-Board) des Rechners befindet sich BIOS
- BIOS – Basic Input/Output System dient der Initialisierung des Systems
- Nach Einschalten beginnt CPU zu laufen
  - Bei mehreren Prozessoren wird dynamisch eine CPU ausgewählt
  - Weitere CPUs werden angehalten und erst später vom Betriebssystem aktiviert
  - CPU läuft im Real Mode
    - Nur 1 MB adressierbar
    - Kein Speicherschutz
    - Keine Unterscheidung von Privilegien



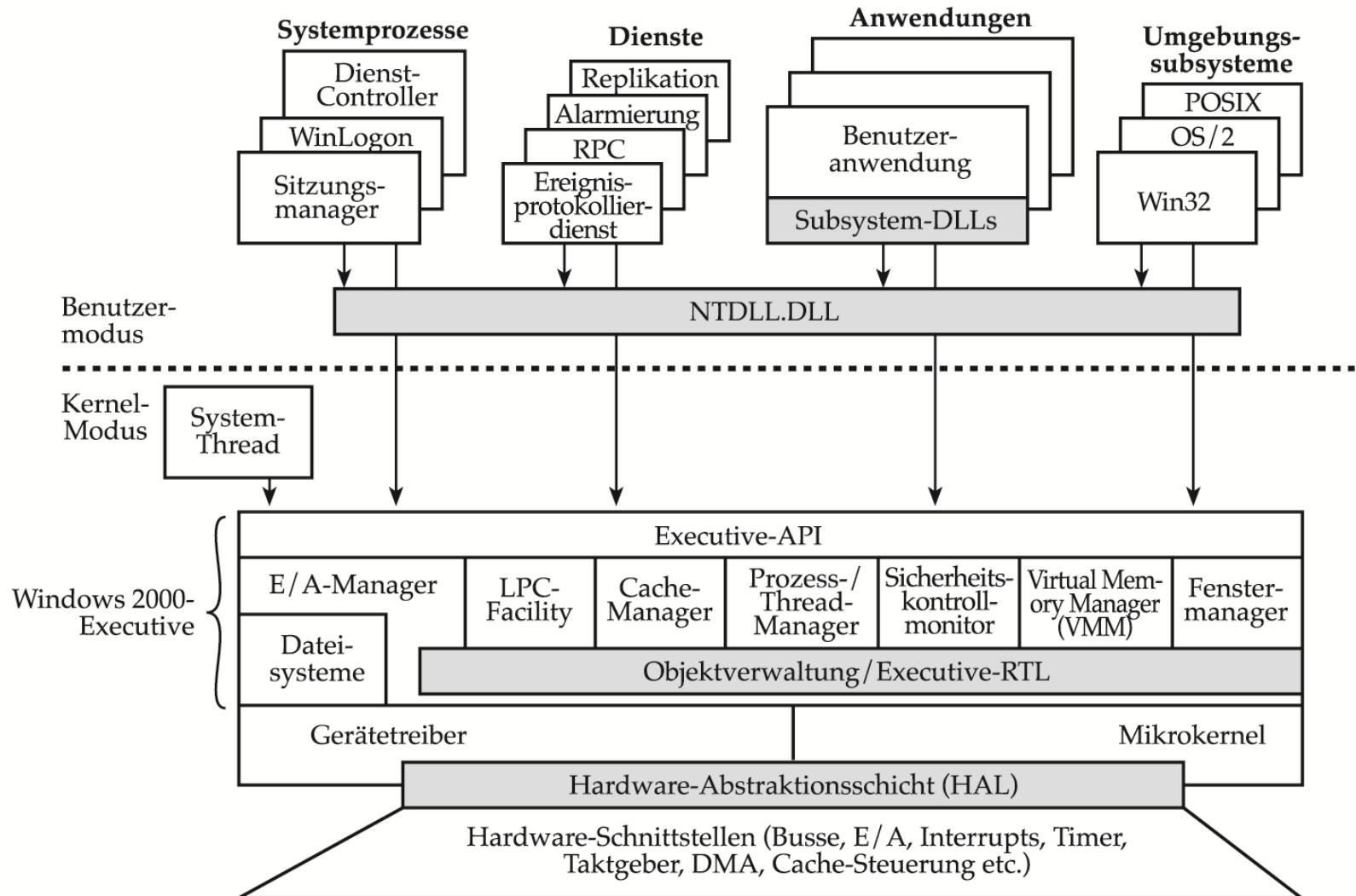
# ABLAUF BEIM BOOTING

- CPU beginnt mit Ausführung des BIOS Code
  - Power-On Self-Test (eine Mischung aus Test und Initialisierung)
  - Enumeratorierung weiter Hardware
  - Initialisierung von Controlern durch Ausführung von Option ROM Code
  - Suche nach einem Boot-Gerät
    - Reihenfolge der Suche konfigurierbar
    - Boot-Gerät enthält MBR im ersten/nullten Sektor
    - Boot-Gerät erkennbar durch MBR-Signatur 0x55AA am Ende vom ersten/nullten Sektor
  - Kopieren des MBR von Boot-Gerät an Speicheradresse 0x7c00
- CPU beginnt mit Ausführung des MBR-Code an Adresse 0x7c00
  - Typischer Microsoft MBR-Code liest Partitionstabelle des MBR um aktive Partition zu ermitteln
  - Laden des Bootsektors der aktiven Partition in den Speicher

# ABLAUF BEIM BOOTING

- CPU beginnt mit Ausführung von geladenem Bootsektor-Code
  - Realisiert minimalen Dateisystem-Treiber
  - Code lädt Datei C:\Bootmgr (früher ntldr)
- CPU führt geladene Datei aus
  - liest Konfiguration aus C:\Boot\BCD (früher boot.ini)
  - schaltet CPU in den Protected Mode um
  - initialisiert Subsysteme des Betriebssystems
  - Lädt Betriebssystemkern (C:\Windows\System32\ntoskrnl.exe) und führt ihn aus
    - Initialisiert verbleibende Subsysteme
    - erste Nutzerprozesse (Anwendungsprogramme) werden angelegt

# LAUFENDES SYSTEM (HIER WINDOWS)



# WIE GREIFEN ANWENDUNGSPROGRAMME AUF HARDWARE ZU

- Anwendungsprogramme laufen im Benutzer-Modus (Ring 3) mit geringen Privilegien
  - I/O zählt zu privilegierten Operationen/Instruktionen
- Betriebssystem im Kernelmodus (Ring 0) stellt Anwendungen Dienste zu Verfügung
  - Nutzung über definierte Schnittstelle: Systemcall
- Anwendungsprogramm ruft Systemcall (typischerweise in Bibliotheksfunktion gekapselt)
- Auslösen eines Interrupts um Kernel die Nutzung eines Systemcalls anzugeben
- Interrupt setzt CPU in Kernel-Modus zur Behandlung des Interrupts durch vorgesehene Funktion (Systemcall-Umsetzung)
- Rückkehr von Interrupt-Behandlung setzt CPU zurück in Benutzer-Modus

21

# WIE ERHÄLT DER BETRIEBSSYSTEMKERN DIE AUSFÜHRUNGSKONTROLLE?

- Durch das Betriebssystem sind regelmäßig Aufgaben zu erledigen
- Auch wenn Anwendungsprogramme laufen
- Timer-Interrupt gibt dem Betriebssystemkern die Kontrolle

22

# FRAGEN?

23

# IT-SICHERHEIT

## 1. INFORMATIONSTECHNIK – COMPUTER UND NETZE

Wintersemester 2015/16

Prof. Dr. Michael Meier  
Informatik IV  
Arbeitsgruppe IT-Sicherheit



Universität Bonn  
E-Mail: mm@cs.uni-bonn.de

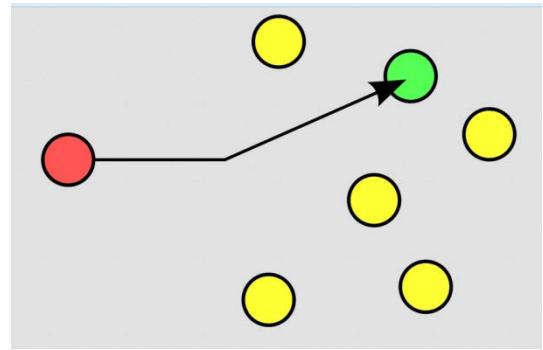
# ÜBERBLICK

- Wiederholung und Festigung grundlegender Kenntnisse
- Vereinfachter Überblick
  - Aufbau und Abläufe von Computersystemen
  - Aufbau und Funktionsweise von Computernetzwerken

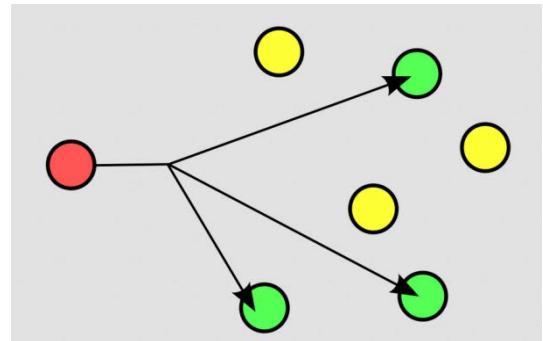
2

# BEGRIFFE: KOMMUNIKATIONSFORMEN / ADRESSIERUNGSAARTEN

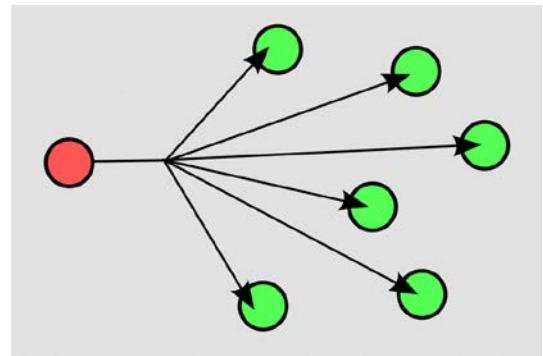
- Unicast-Kommunikation
  - einer kommuniziert mit einen



- Multicast-Kommunikation
  - einer kommuniziert mit viele (eine Gruppe)



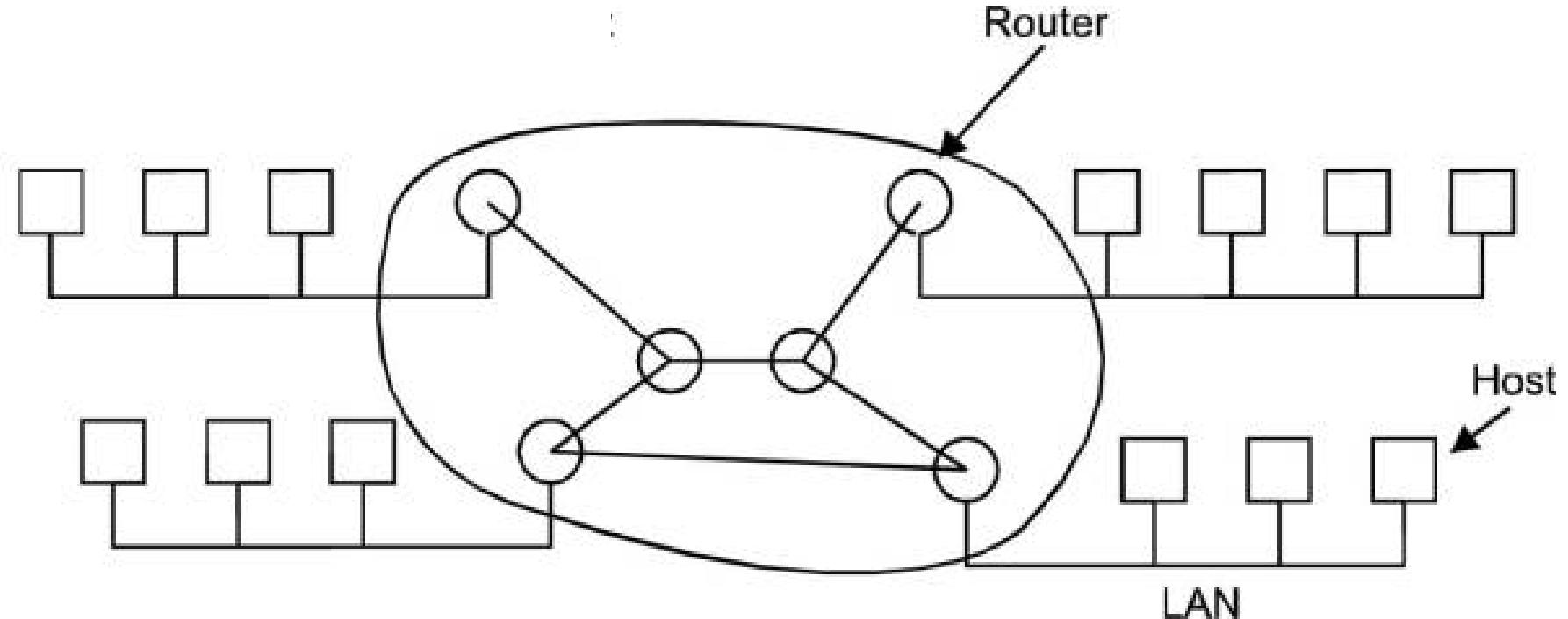
- Broadcast-Kommunikation
  - einer sendet an alle



3

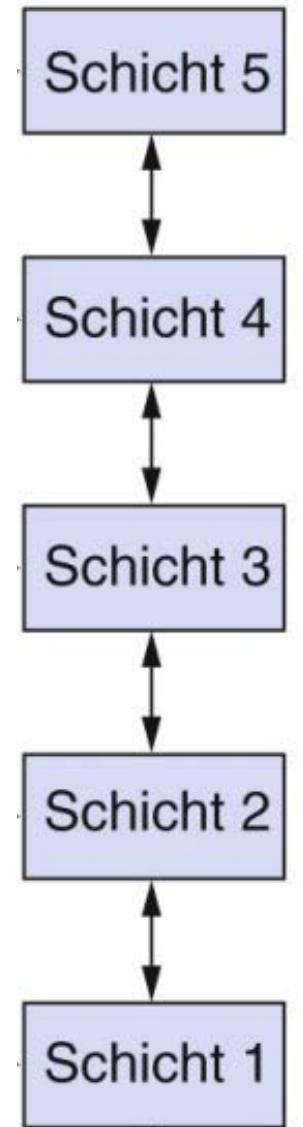
# NETZE UND DEREN KOPPLUNG

4

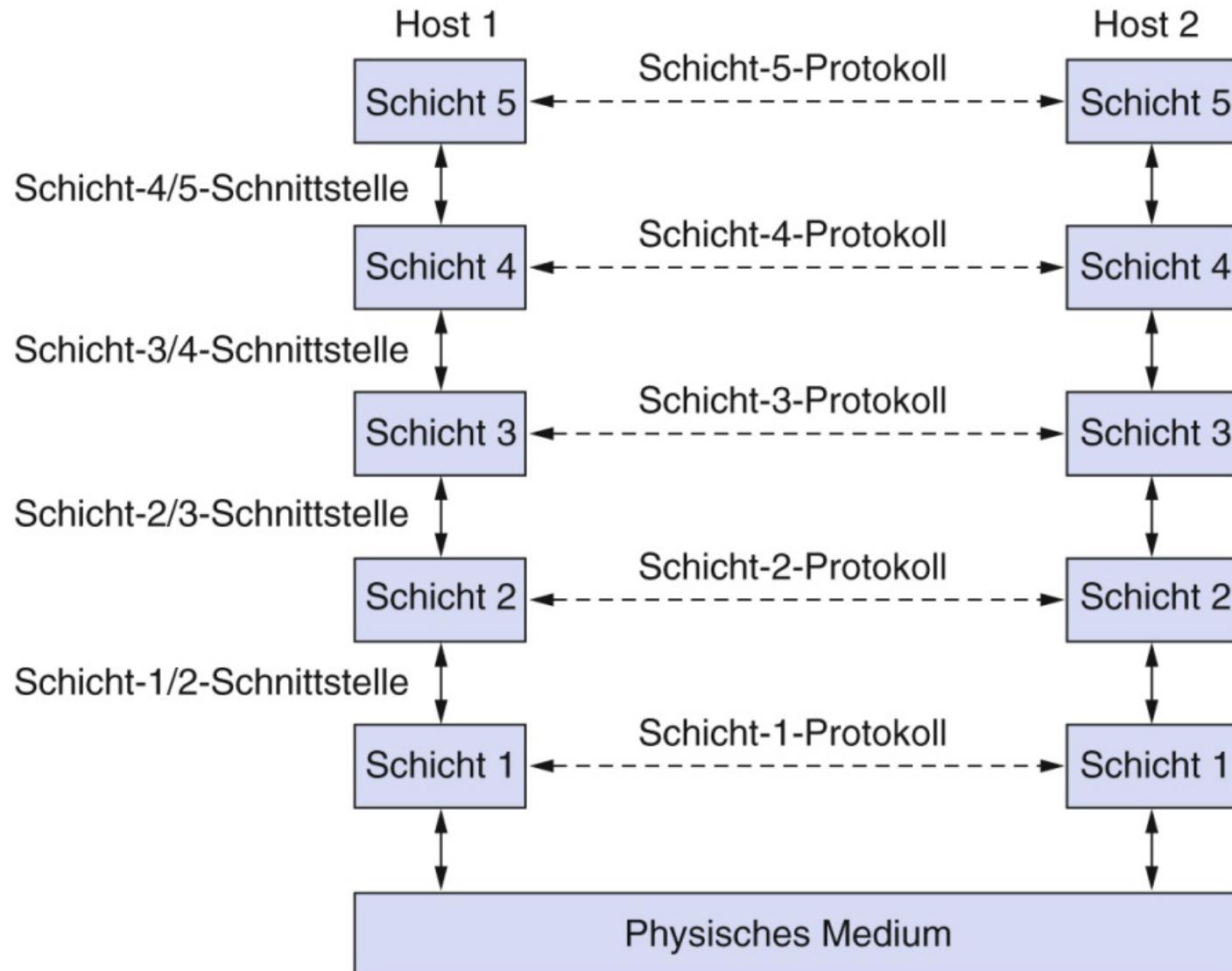


# PROTOKOLLHIERARCHIEN/NETZWERKSTACKS

- Warum verschiedene Schichten?
  - ⇒ Komplexitätsbeherrschung:  
Funktionen/Aufgaben in Schichten aufgeteilt
- Schicht n bei Host 1 redet mit Schicht n bei Host 2
- Regeln und Konventionen des „Gesprächs“ auf Schicht n bilden das **Protokoll** der Schicht n
- In Wirklichkeit kein Datenaustausch auf Schicht n, sondern **Daten und Steuerinformation** werden an darunterliegende Schicht n-1 weitergereicht, bis unterste Schicht erreicht ist.
- Unter Schicht 1 liegt das **physische Medium** über das Kommunikation stattfindet
- **Schnittstellen** zwischen angrenzenden Schichten definieren welche **Dienste** und Funktionen die untere (n-1) der oberen Schicht (n) zur Verfügung stellt

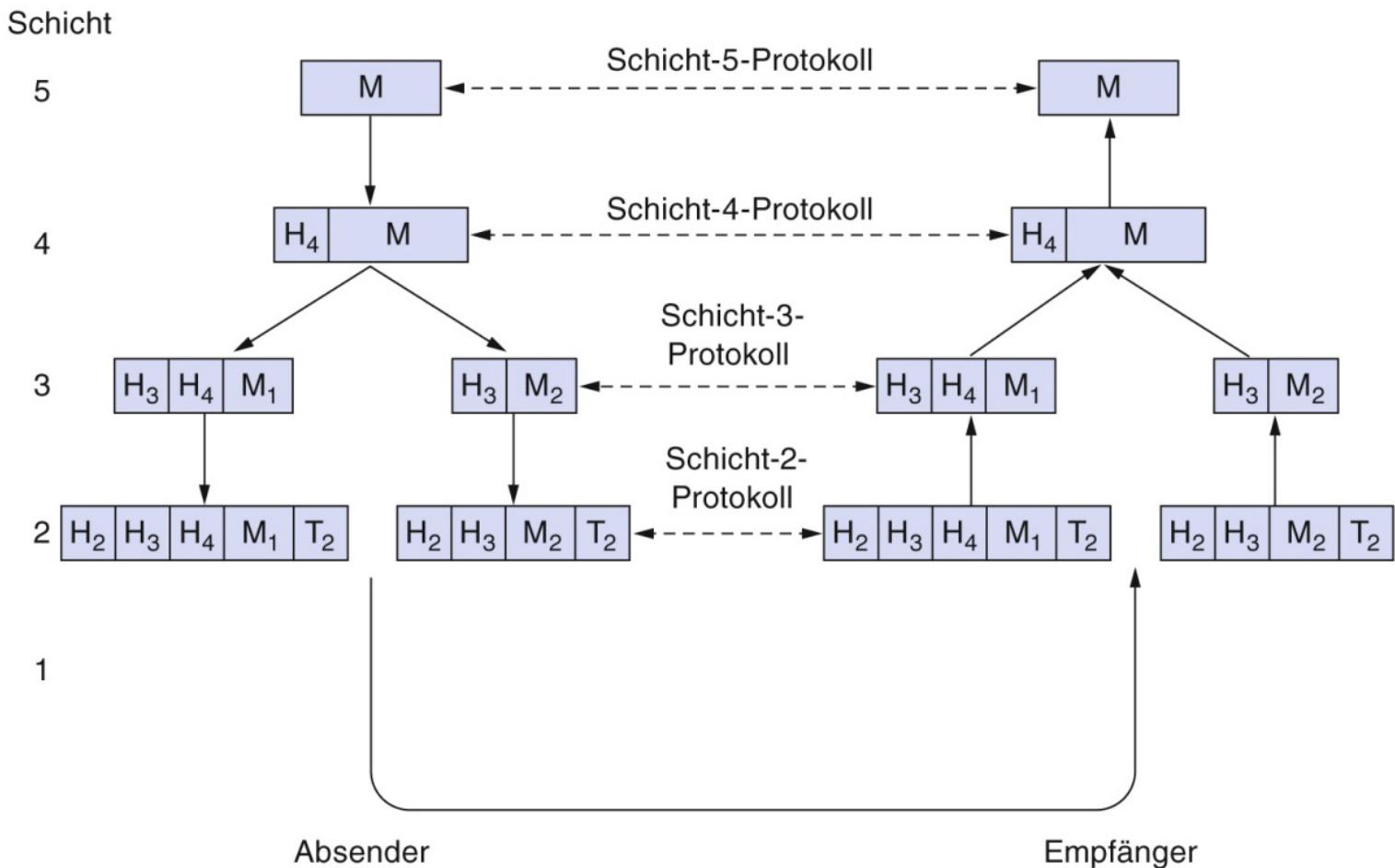


# SCHICHTEN, PROTOKOLLE, SCHNITTSTELLEN



# SCHICHTEN, PROTOKOLLE, SCHNITTSTELLEN

- Konzeptionell horizontale Kommunikation in jeder Schicht:  
SendenAnAndereSeite oder HoleVonAndererSeite
- Tatsächlich kommunizieren die Prozeduren mit der darunterliegenden Schicht



# DESIGNASPEKTE / AUFGABEN (AUSWAHL)

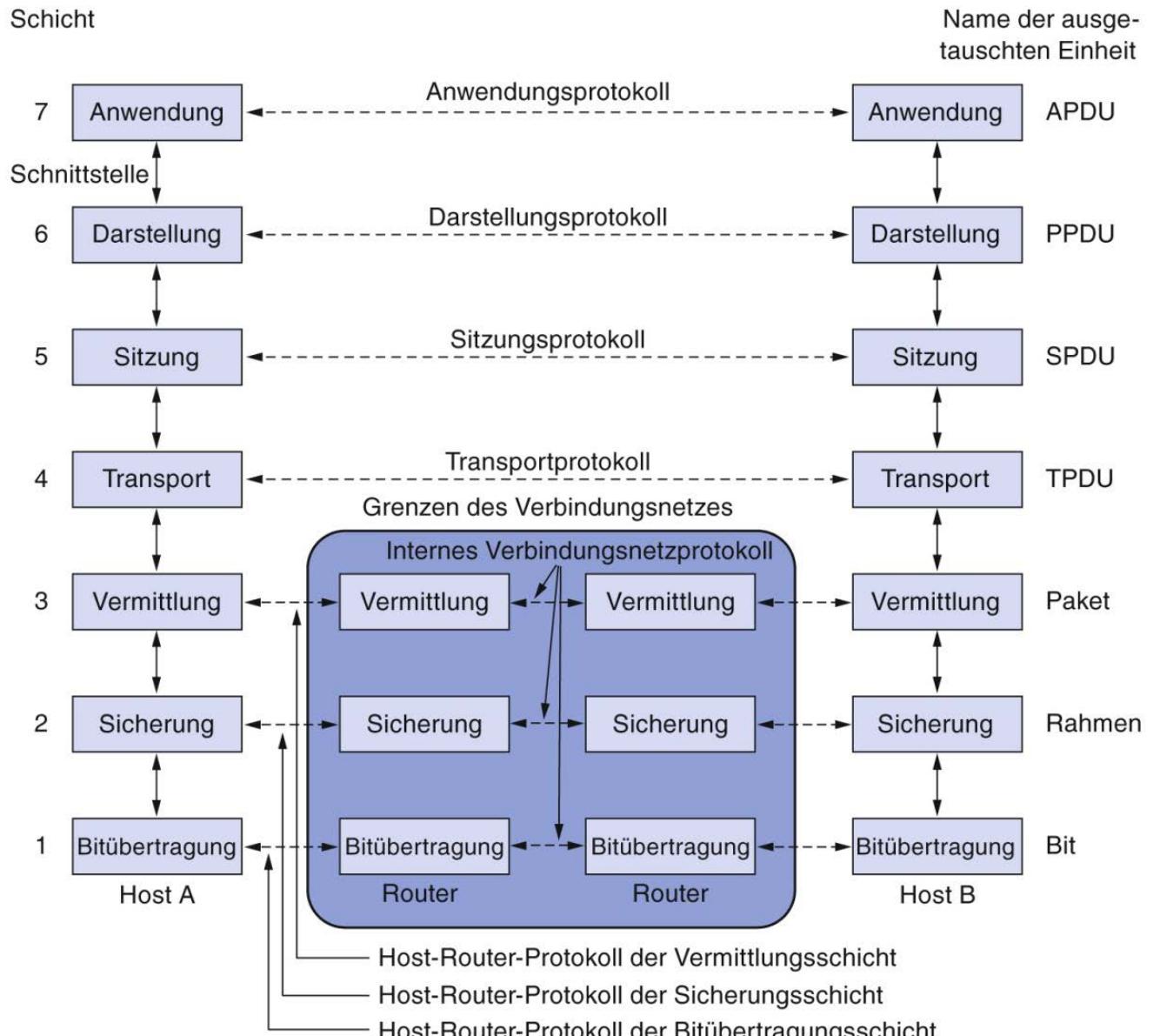
- Verbindungsaubau
- Adressierung (auf jeder Schicht)
- Datentransfer-Regeln:
  - Simplex, Halbduplex, Vollduplex
- Fehlerüberwachung: physische Medien sind nicht perfekt
- Reihenfolgeerhaltung: Nummerierung und Sortierung
- Geschwindigkeitsvereinbarung
- Größenbeschränkungen für Nachrichten: zerstückeln und zusammensetzen
- Optimierungen, gemeinsame Nutzung: Multiplexen und Demultiplexen
- Mehrere Kommunikationswege: Wegwahlentscheidung

# VERBINDUNGSORIENTIERTE UND VERBINDUNGSLOSE DIENSTE

- Zuverlässigkeit: Richtige Reihenfolge mit Empfangsbestätigung
  - Macht Aufwand und kostet Zeit
- Unzuverlässigkeit: keine garantierte Reihenfolge und Verluste möglich
  - Macht weniger Aufwand und kostet weniger Zeit

	Dienst	Beispiel
verbindungs-orientiert	Zuverlässiger Nachrichtenstrom	Folge von Seiten
	Zuverlässiger Bytestrom	Download eines Films
	Unzuverlässige Verbindung	Voice-over-IP
verbindungs-los	Unzuverlässiges Datagramm	Junk-E-Mail
	Bestätigtes Datagramm	Textnachrichten
	Anforderung/Antwort	Datenbankabfrage

# OSI-REFERENZMODELL



# SCHICHTEN DES OSI-REFERENZMODELLS

(OSI – Open System Interconnection)

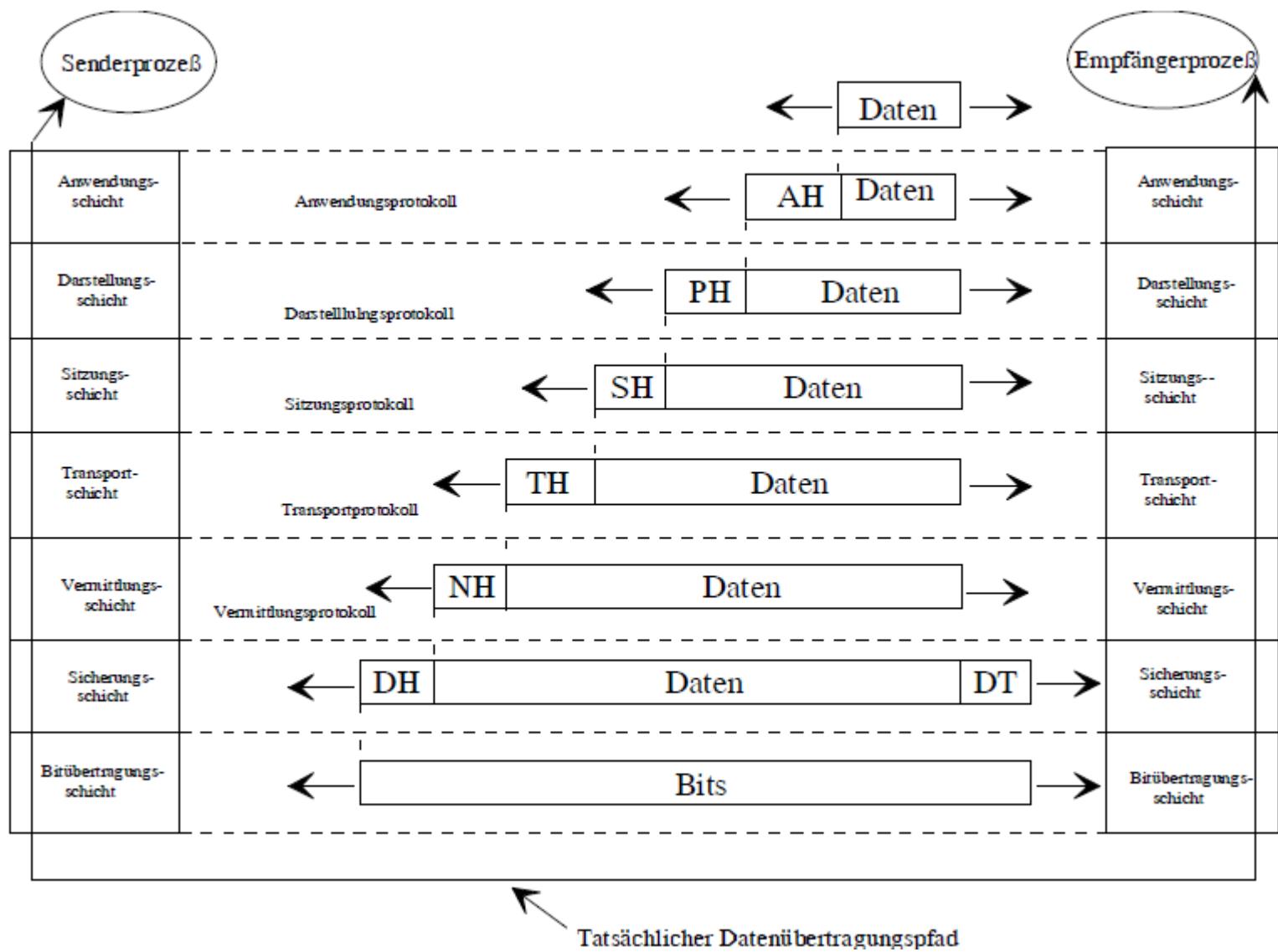
- 1 - Bitübertragungsschicht (Physical Layer)
  - Bits auf physischem Medium
- 2 - Sicherungsschicht (Data Link Layer)
  - Teilt Daten in Rahmen (Data Frames) auf, markiert Rahmengrenzen
  - Verarbeitet Bestätigungsrahmen
  - Rahmen und Bestätigungsrahmen können verloren gehen
  - Erneute Übertragung; mögliche Duplikate
  - In Broadcast-Netzen: Steuerung des Zugriffs auf das gemeinsame Medium, durch MAC-Teilschicht
- 3 - Vermittlungsschicht (Network Layer)
  - Wegwahlentscheidung
  - Größenanpassung

11

# (ENDE-ZU-ENDE-) SCHICHTEN DES OSI-MODELLS

- 4 - Transportschicht (Transport Layer)
  - Mehrprogrammbetrieb: Welche Nachricht zu welcher Verbindung
  - Flusssteuerung / Geschwindigkeitsvereinbarung (zwischen Hosts)
- 5 - Sitzungsschicht (Session Layer)
  - Synchronisation und Ähnliches
- 6 - Darstellungsschicht (Presentation Layer)
  - Häufige Funktionen dem Benutzer abnehmen
  - Standardkodierungen (z.B. ASCII, Unicode)
- 7 - Verarbeitungsschicht/Anwendungsschicht (Application Layer)
  - Vielzahl häufig benötigter Protokolle
  - Dateitransfer
  - E-Mail
  - ...

12



# TCP/IP vs. OSI-RERERENZMODELL

## ■ OSI

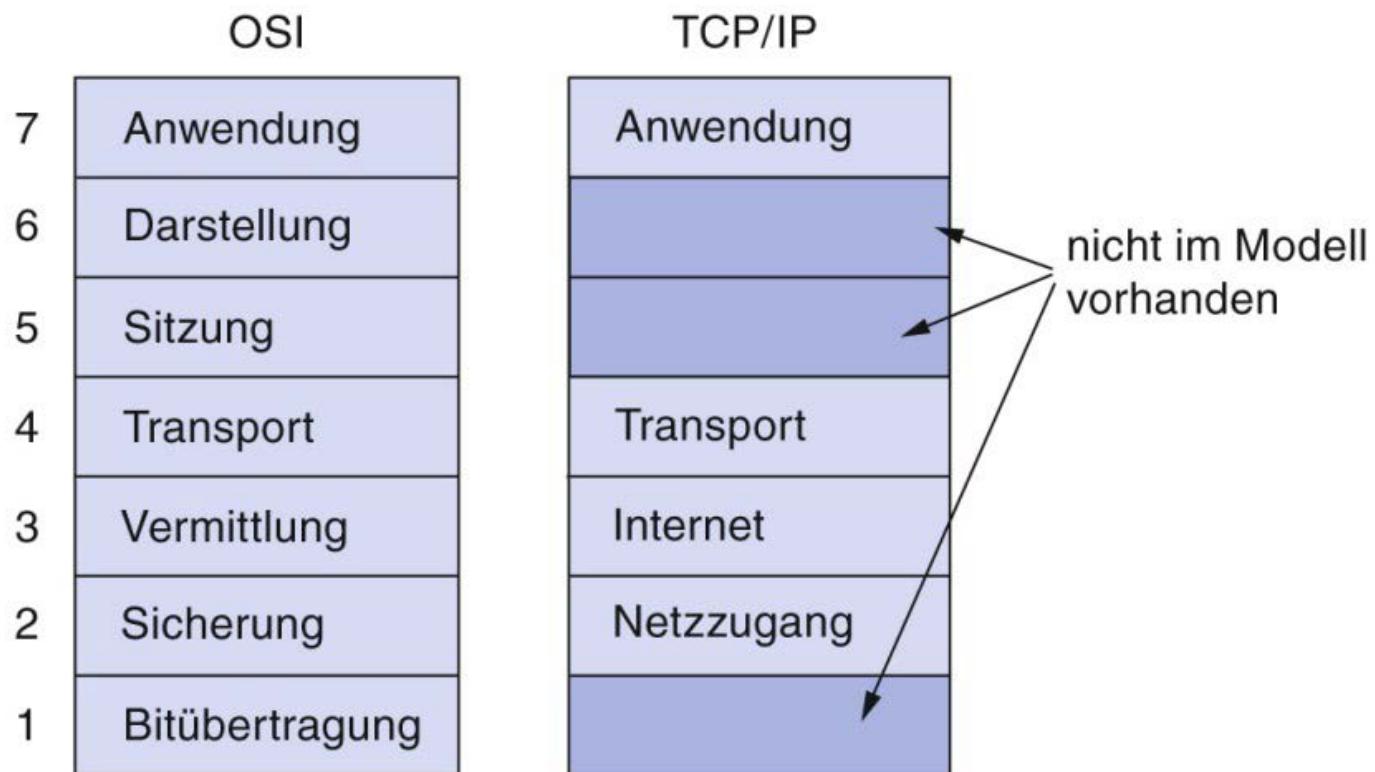
- Erst wurde das Referenzmodell entwickelt  
dann die Protokolle
- Hat sich nicht durchgesetzt; in der Praxis kaum anzutreffen
- Wird dennoch zur Betrachtung/Diskussion von Netzen verwendet

14

## ■ TCP/IP

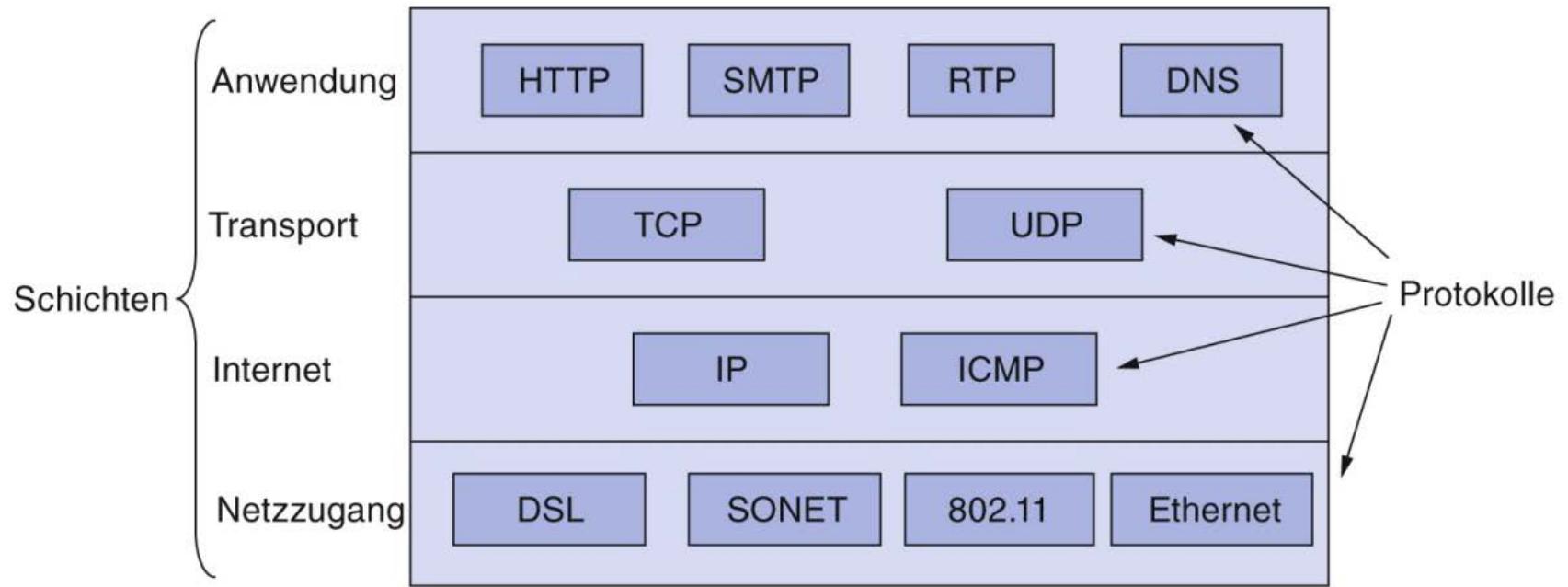
- Erst wurden Protokolle entwickelt  
dann Referenzmodell

# TCP/IP-REFERENZMODELL



15

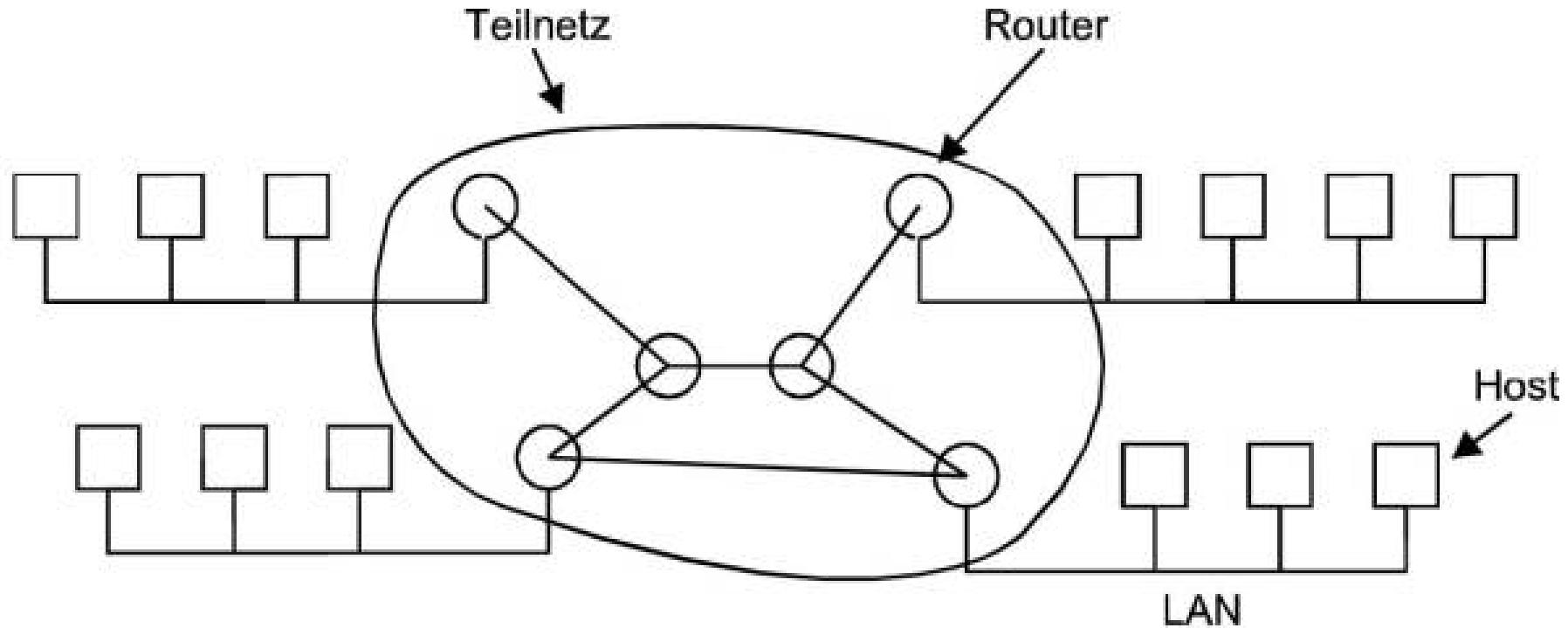
# PROTOKOLLE UND NETZWERKE IM TCP/IP-MODELL



- SONET – Synchronous Optical Network
- 802.11 = WLAN

# NETZWERKE

- Punkt-Zu-Punkt-Netzwerk von Routern
- Typischerweise Broadcast-Netzwerk im LAN



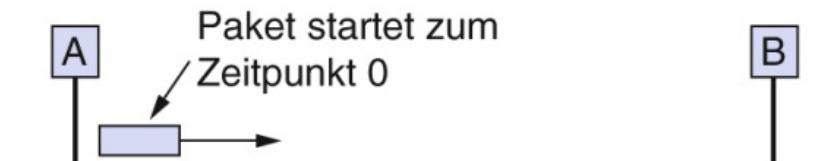
## ■ Broadcast-Netze

- Alle angeschlossenen Hosts nutzen gemeinsames physikalisches Medium (Broadcast-Kanal), das jeder beschreiben und lesen kann
- Gleichzeitiges Senden (Kollision) führt zur „Verstümmelung“ der übertragenen Daten

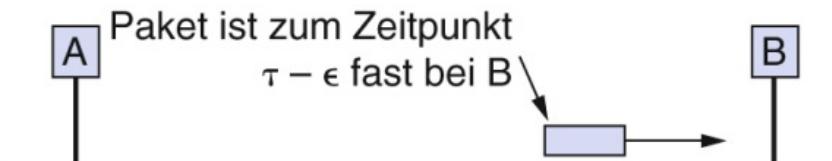
## ■ Mehrfachzugriffsprotokolle regeln den Zugriff

- Medium Access Control (MAC)
- Carrier Sense Multiple Access (CSMA)
  - Hören ob der Kanal frei ist, erst dann senden
  - Aufgrund von Ausbreitungsverzögerungen treten dennoch Kollisionen auf

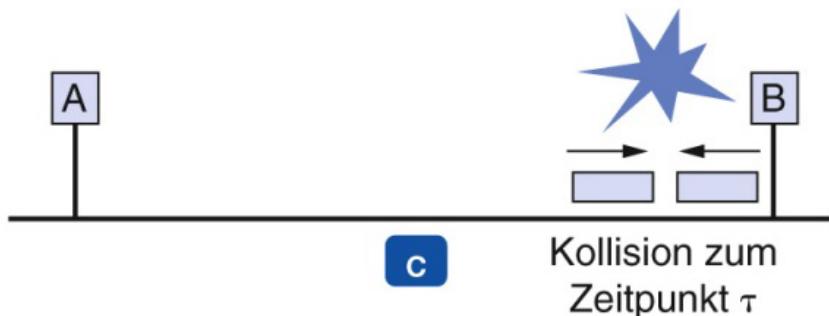
# NETZZUGANGSSCHICHT



a



b



c

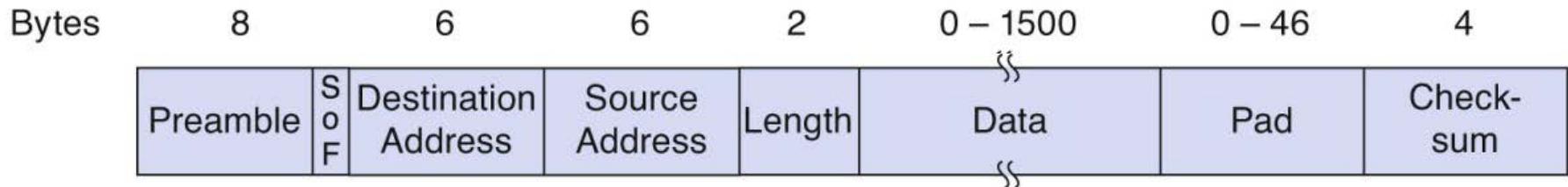


d

- Kollisionserkennung (Collision Detection (CD))
  - Abbruch der laufenden Übertragung (die eh verstümmelt wird)
  - Erneute Übertragung

# NETZZUGANGSSCHICHT - ETHERNET

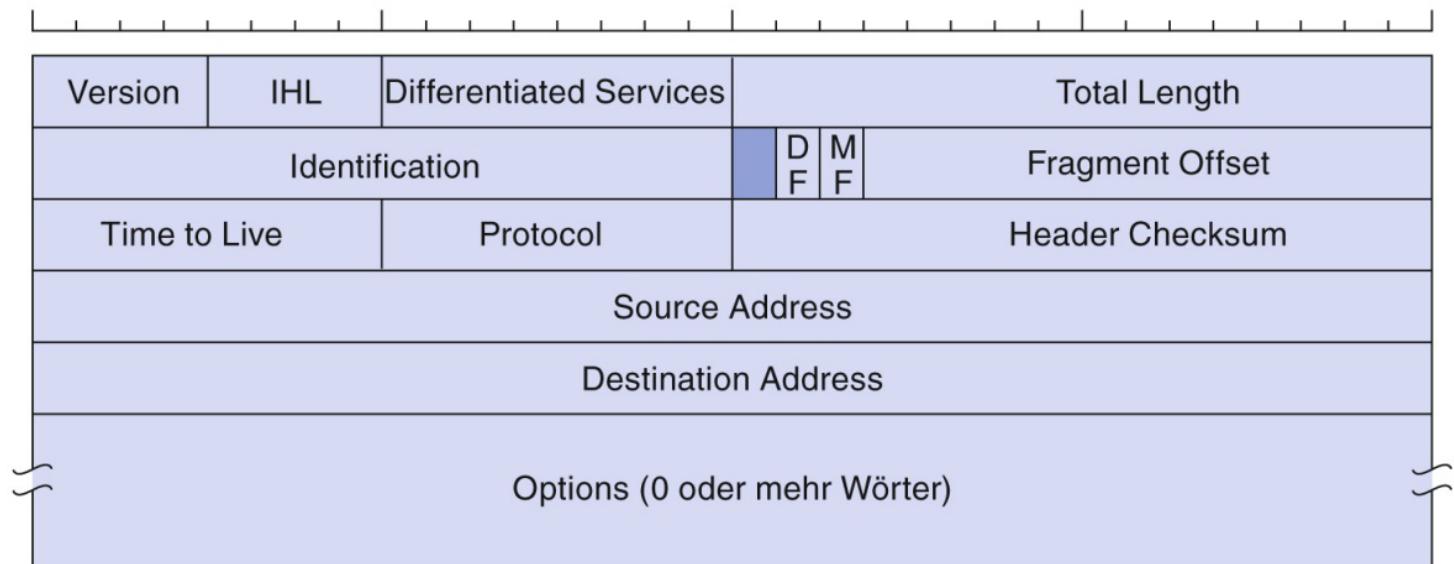
- Arbeitet mit CSMA/CD
- Rahmenformat



- SoF = Start of Frame
- Optionales Pad für evtl. Auffüllen zur Erreichung der Mindestlänge
- auf Netzwerkkarten/-adapters implementiert
  - Hersteller vergeben feste und theoretisch weltweit eindeutige MAC-Adressen/Hardware-Adresse vergeben
    - **00-07-E9-XX-XX-XX**
      - Firma Intel
      - XX-XX-XX: laufende Nummer beim Hersteller Intel
    - **FF-FF-FF-FF-FF-FF**
      - Broadcast-Adresse: alle Geräte im lokalen Netz

# IP – INTERNET PROTOKOLL

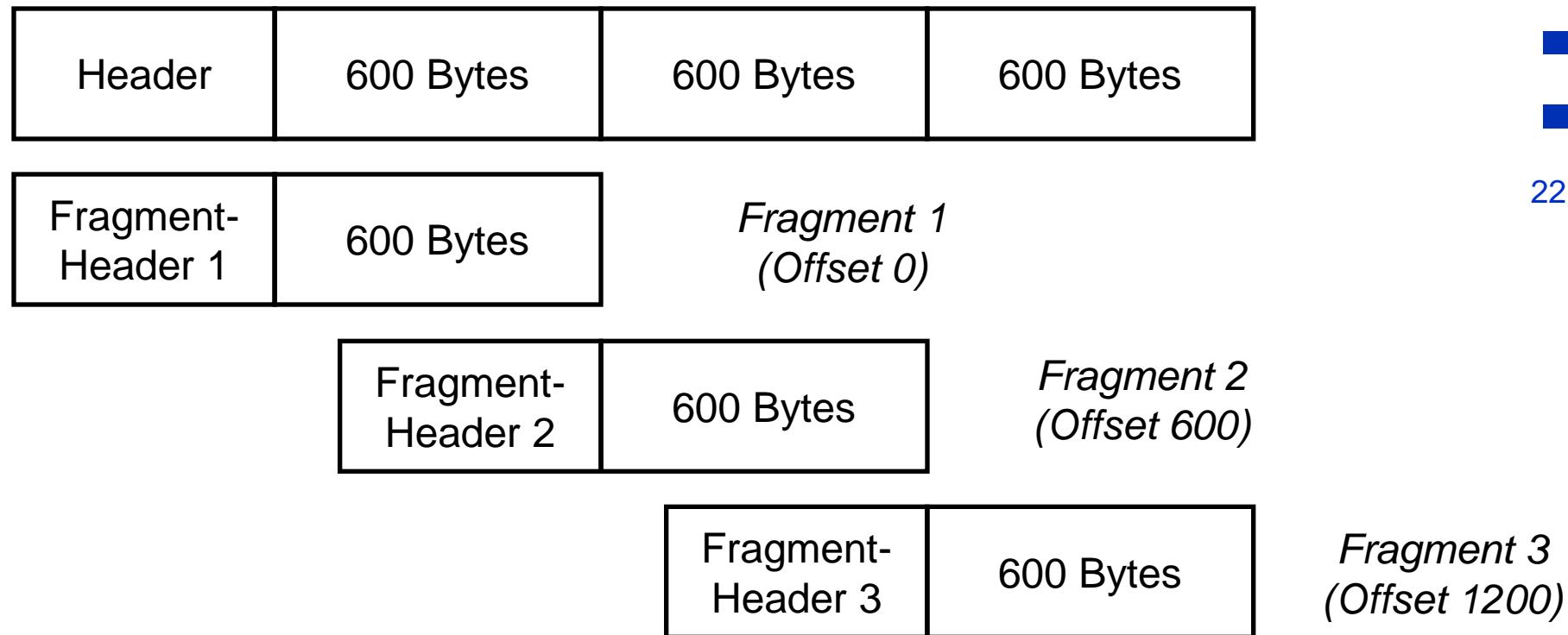
- IP realisiert unzuverlässigen Datagram-Dienst
- IP-Header



- IHL: Header Länge
- Total Length: Länge von Header und Daten, max. 65.535 Byte
- Identification: zwecks Zuordnung gleich für alle Fragmente eines Datagramms
- Protocol: TCP, UDP, ...

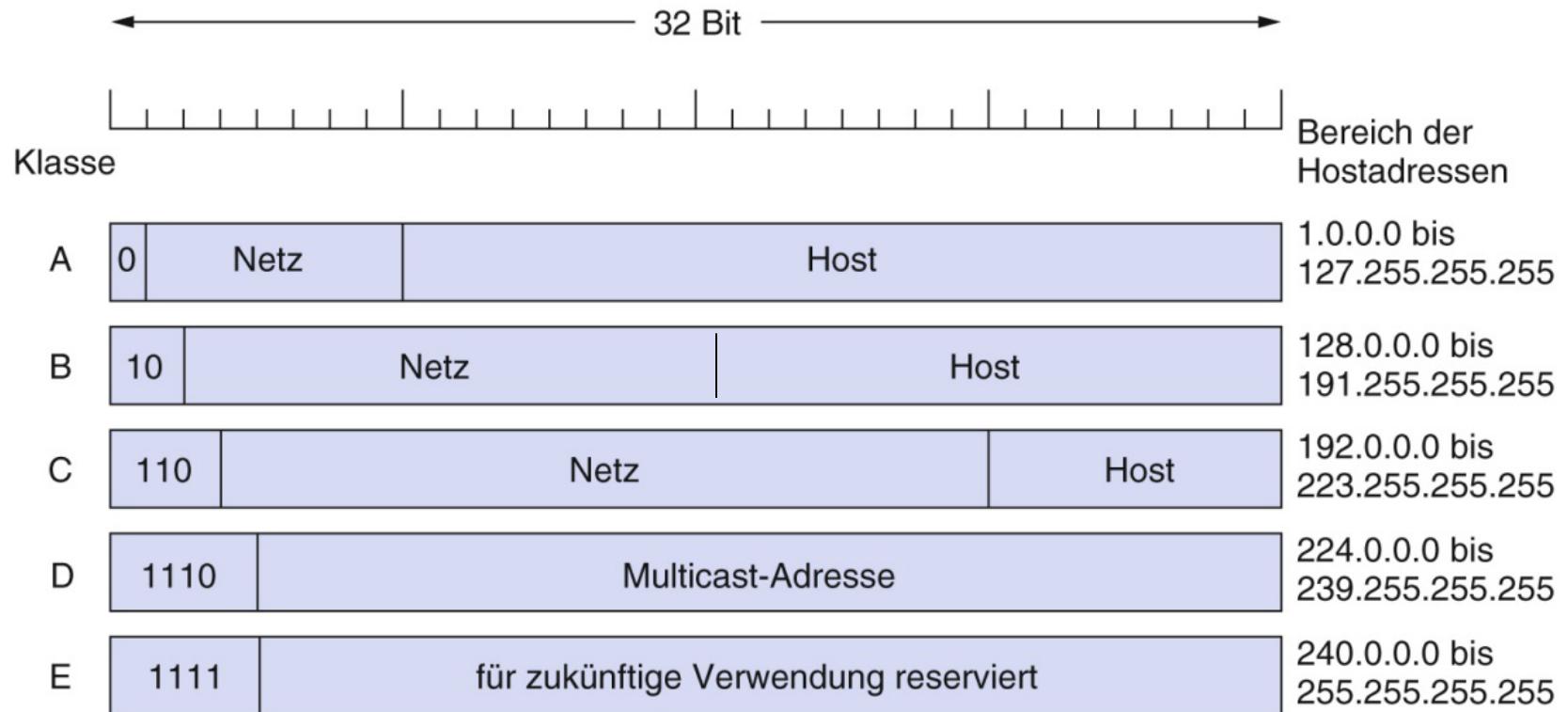
# FRAGMENTIERUNG

- Datagramm ggf. zu groß für Netzzugangsschichtprotokoll
  - MTU – Maximum Transfer Unit



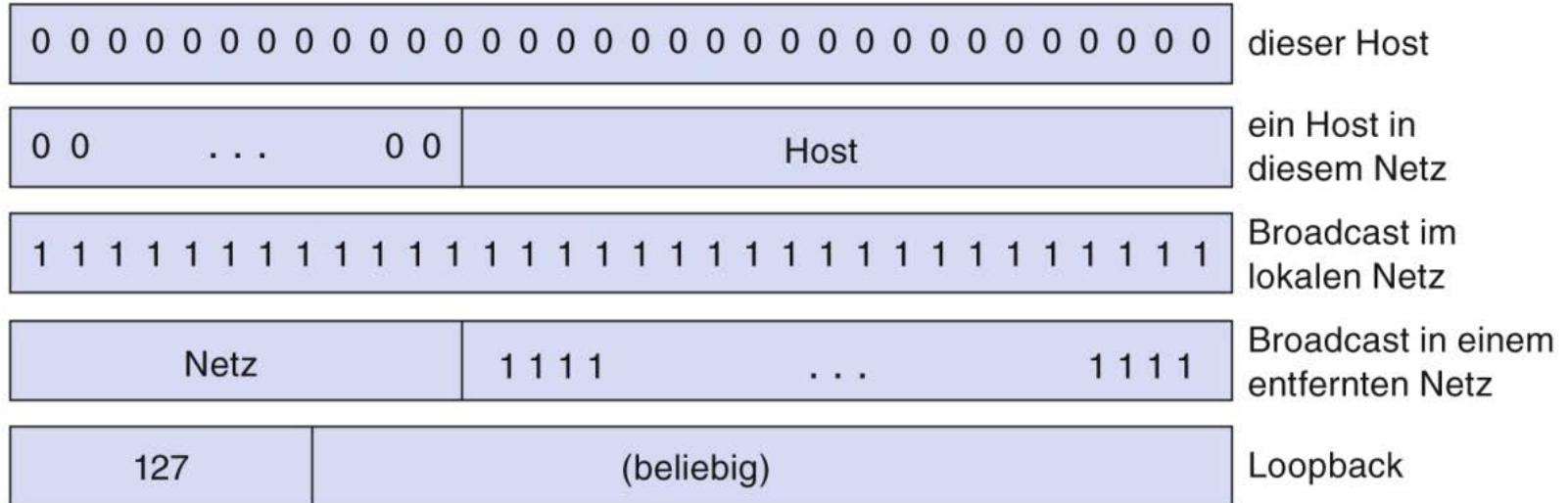
# IP – ADRESSEN

## ■ Adressen und Netzmasken



## ■ IP-Adresse kodiert Host- und Netznummer

# SPEZIELLE IP-ADRESSEN

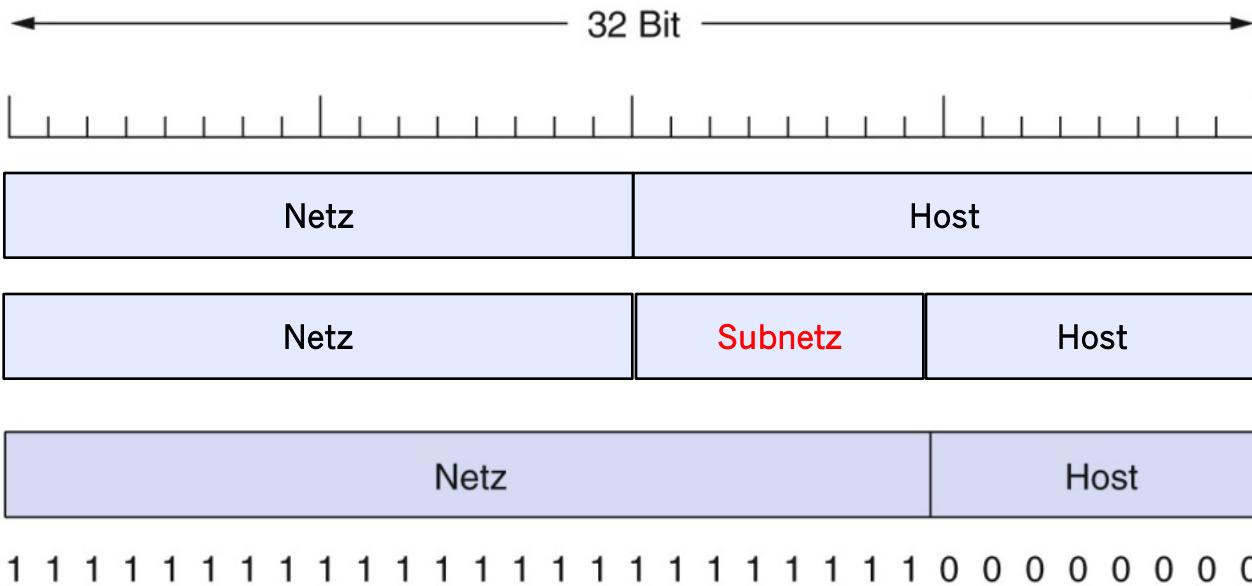


# IP-ADRESSEN

- Netzklassen allein erweisen sich als recht unflexibel

## ⇒ IP-Subnetze

- Netz-interne Unterteilung in mehrere IP-Subnetze durch Vergrößerung des Netzteils einer IP-Adresse
- Die ersten Bits der Host-Nummer werden als Subnetz-Nummer genutzt



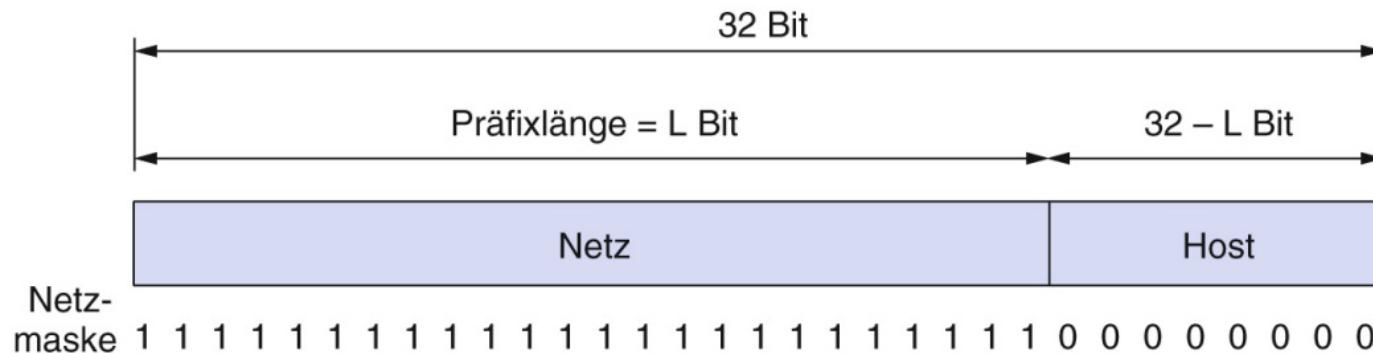
- Netzmaske definiert welche Bits zum (Sub-)Netzteil gehören

# IP-ADRESSEN

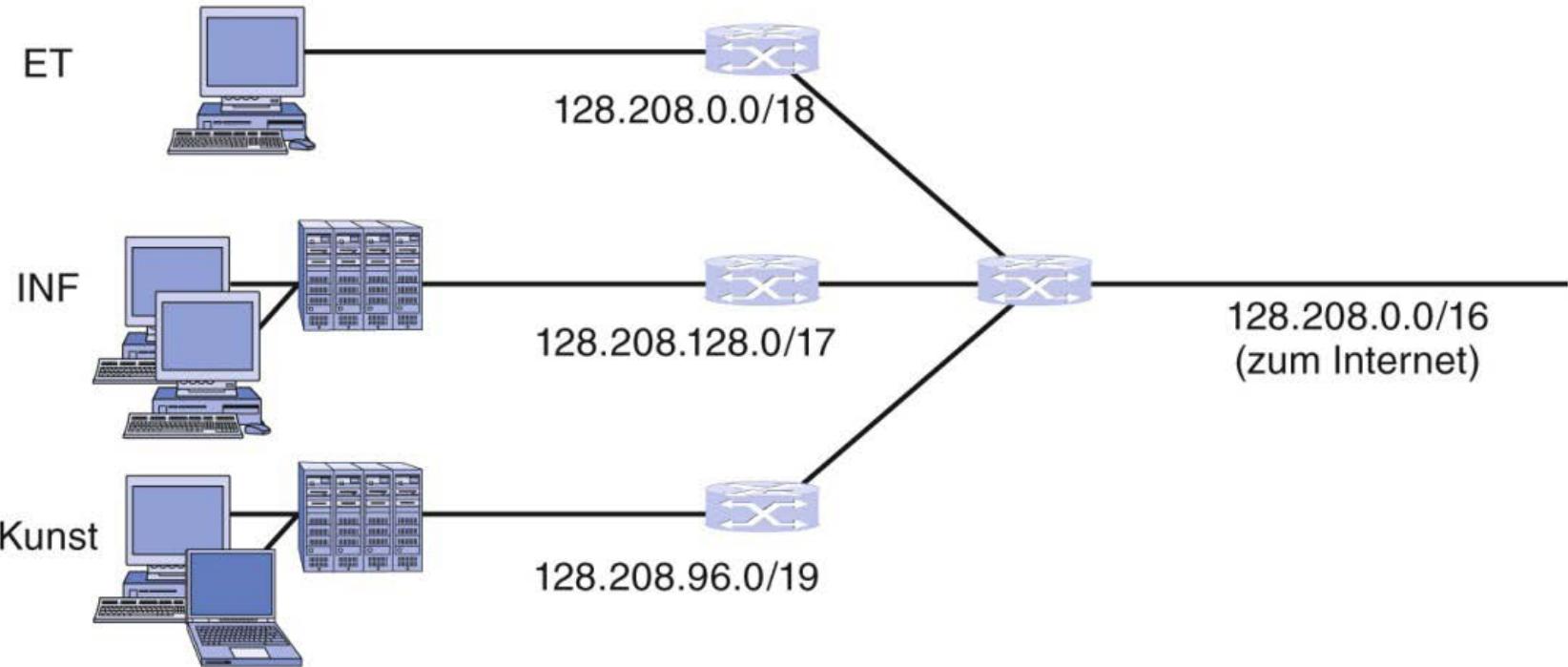
## ■ Classless Inter-Domain Routing (CIDR)

- Effizientere Nutzung des IP-Adressraums
- feste Zuordnung einer IP-Adresse zu einer Adressklasse entfällt, aus der die Länge des Netzteils der IP-Adresse hervorging
- Variable Länge des Netzteils (Netz-Präfix)
  - Festgelegt durch Netzmaske oder Präfixlänge

- CIDR-(Suffix)-Notation für (Sub-)Netze: IP-Adresse/Präfixlänge
- IP-Adresse/24



# SUBNETTING – AUFSPALTEN IN SUBNETZE



27

# ROUTING

## ■ Routingtabelle eines Rechner/Routers

### ■ Spalte Ziel-Adressen

- Netzadressen (IP-Adresse/Präfixlänge)
- Hostadressen im eigenen/lokalen Netz (IP-Adresse)
- DEFAULT: alle anderen Ziele die nicht durch eine anderen Eintrag betroffen sind

### ■ Spalte Weg zum Ziel

- Adresse des nächsten Routers auf dem Weg zum Ziel und Netzwerkkarte über die er erreicht werden kann (für entfernte Ziele)
- Netzwerkkarte über die das Ziel direkt erreicht werden kann (für Ziele im lokalen Netz)

### ■ Typische Routing-Tabelle eines Rechners

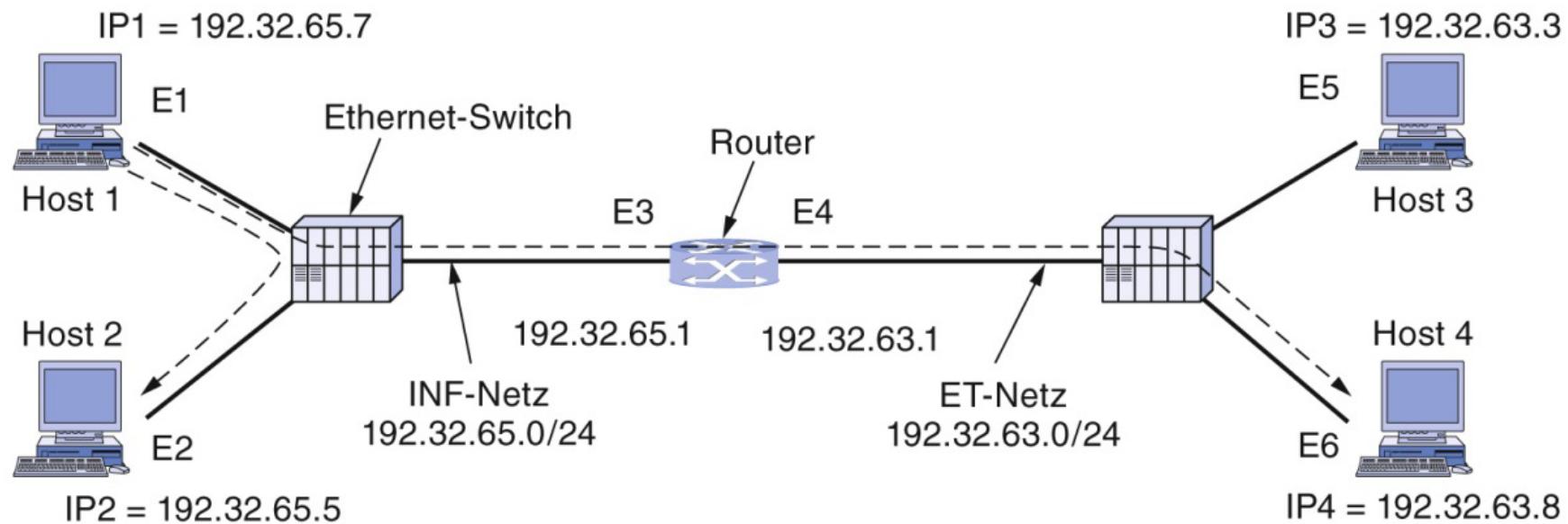
Ziel-Adressen	Weg zum Ziel
IP-Adresse/Präfix	ETH0
Default	Default-Router-IP-Adresse, ETH0

Eigene lokale Netzadresse

Netzwerkkarte

Router im lokalen Netz

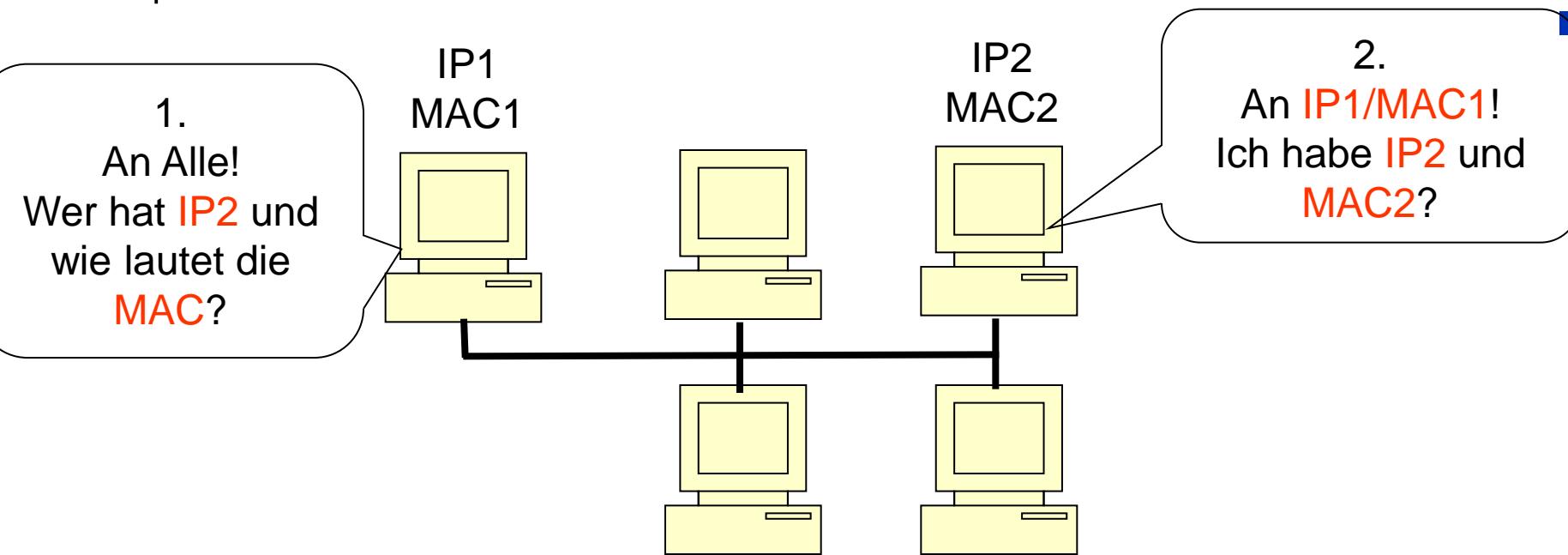
# MAC/ETHERNET- UND IP-ADRESSEN



Rahmen	Quell-IP	Quell-Ethernet	Ziel-IP	Ziel-Ethernet
Host 1 zu 2 im INF-Netz	IP1	E1	IP2	E2
Host 1 zu 4 im INF-Netz	IP1	E1	IP4	E3
Host 1 zu 4 im ET-Netz	IP1	E4	IP4	E6

# ADDRESS RESOLUTION PROTOKOLL (ARP)

- Woher kennen der Rechner zu einer IP-Adresse die MAC-Adresse?
  - ARP löst IP-Adressen in die zugehörigen MAC-Adressen auf
    - Broadcast (FF-FF-FF-FF-FF-FF) an alle: Wer hat die IP-Adresse?
    - Inhaber der IP-Adresse antwortet an den Absender
- Beispiel: Rechner mit IP1 möchte an Rechner mit IP2 senden



- Rechner speichert Zuordnung IP-MAC nur zeitweilig

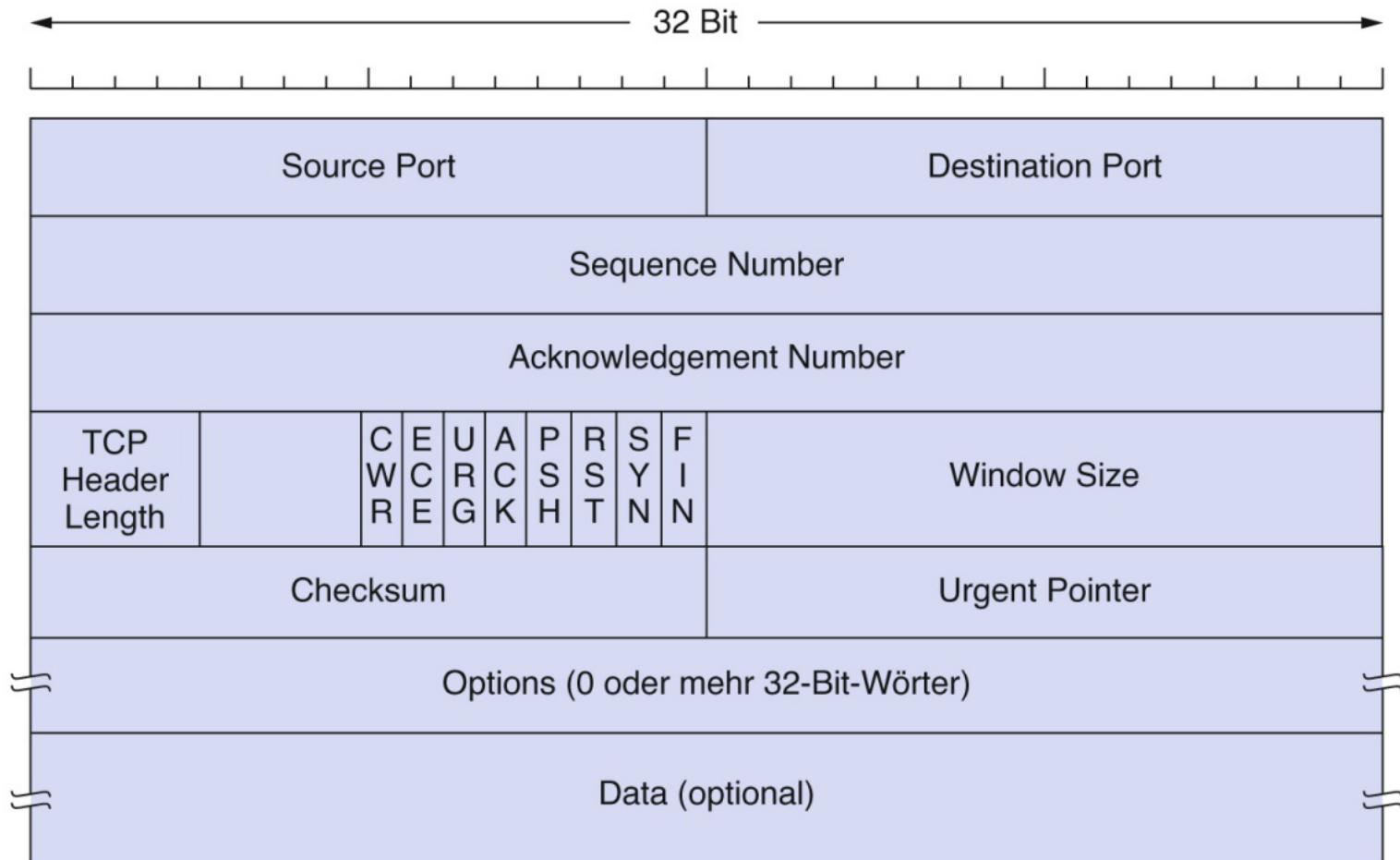
# TRANSPORTSCHICHT – TRANSMISSION CONTROL PROTOCOL (TCP)

- Zuverlässige Byte-Strom-Übertragung von Ende-zu-Ende
- Sender- und Empfänger-Endpunkte werden als **Sockets** bezeichnet
- Socket charakterisiert durch IP-Adresse und Portnummer

31

Port	Protokoll	Verwendung
20, 21	FTP	Dateiübertragung
22	SSH	Fern-Login, Ersatz für Telnet
25	SMTP	E-Mail
80	HTTP	World Wide Web
110	POP-3	Fernzugriff auf E-Mail
143	IMAP	Fernzugriff auf E-Mail
443	HTTPS	Sicheres Web (HTTP über SSL/TLS)
543	RTSP	Mediaplayer-Steuerung
631	IPP	Gemeinsame Druckernutzung

# TCP-HEADER



# TCP-VERBINDUNGSaufbau

■

■

■

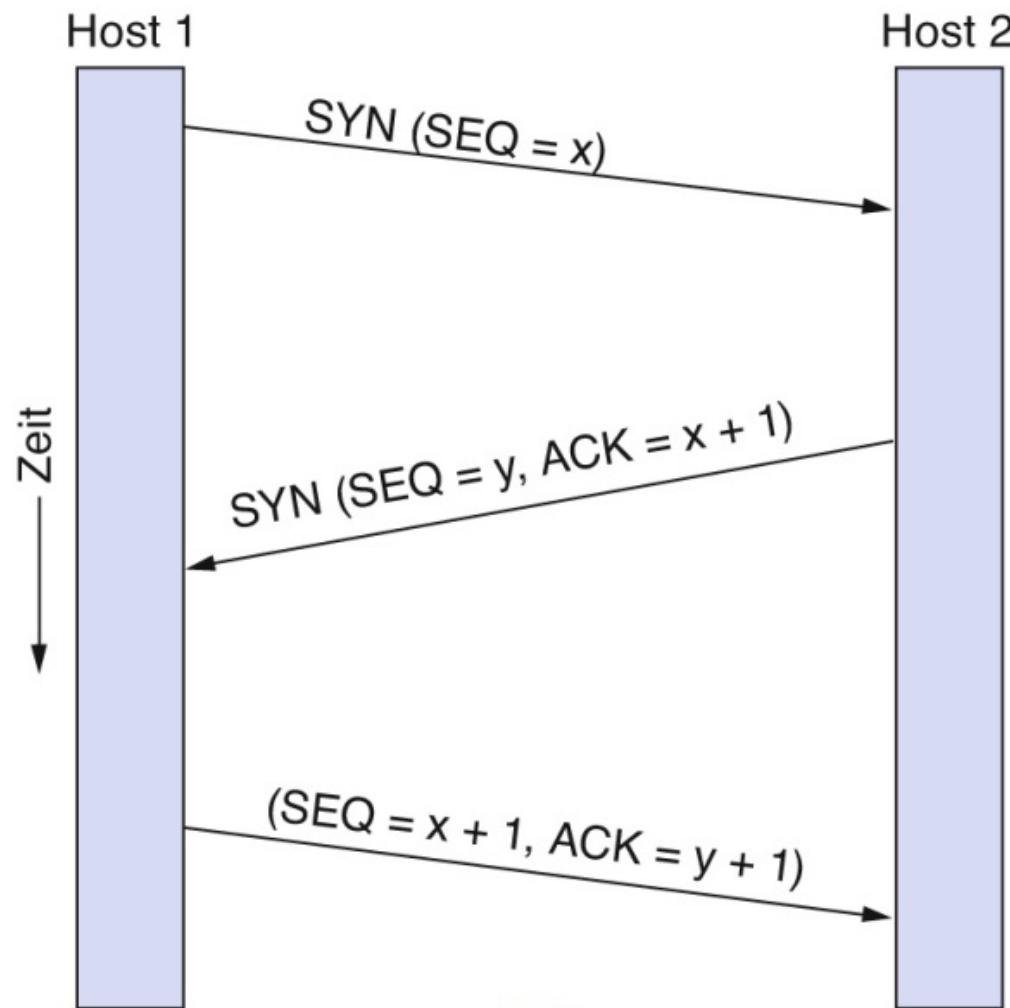
■

■

■

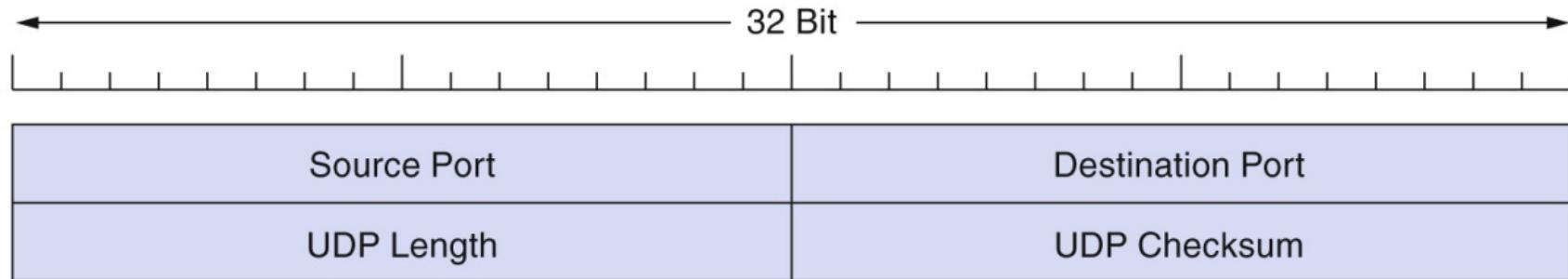
■

33



# TRANSPORTSCHICHT – USER DATAGRAM PROTOKOLL (UDP)

- Verbindungslose, unzuverlässige Ende-zu-Ende-Übertragung von Datagrammen
- UDP Header



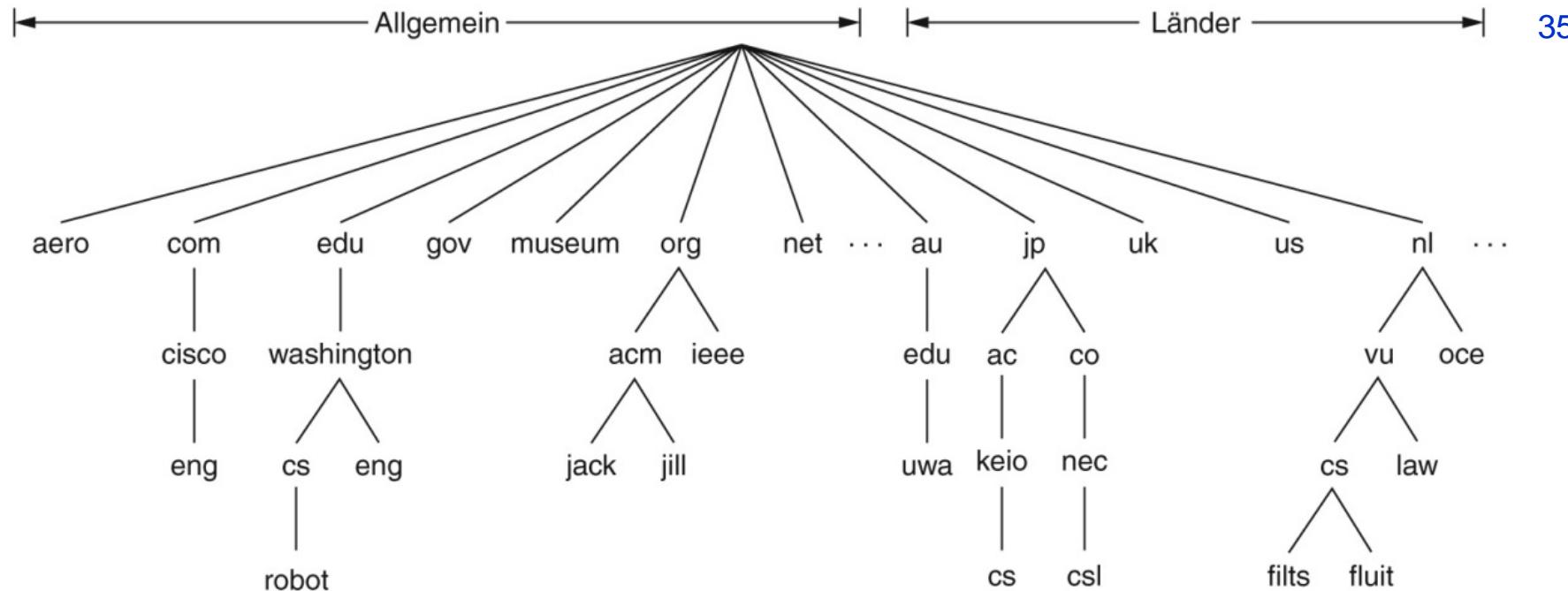
# ANWENDUNGSSCHICHT

## ■ Domain Name System (DNS)

- Textuelle Rechnernamen statt IP-Adressen
- Verwaltung sich ändernder Namen und deren Zuordnung zu IP-Adressen

## ■ DNS Namensbereiche

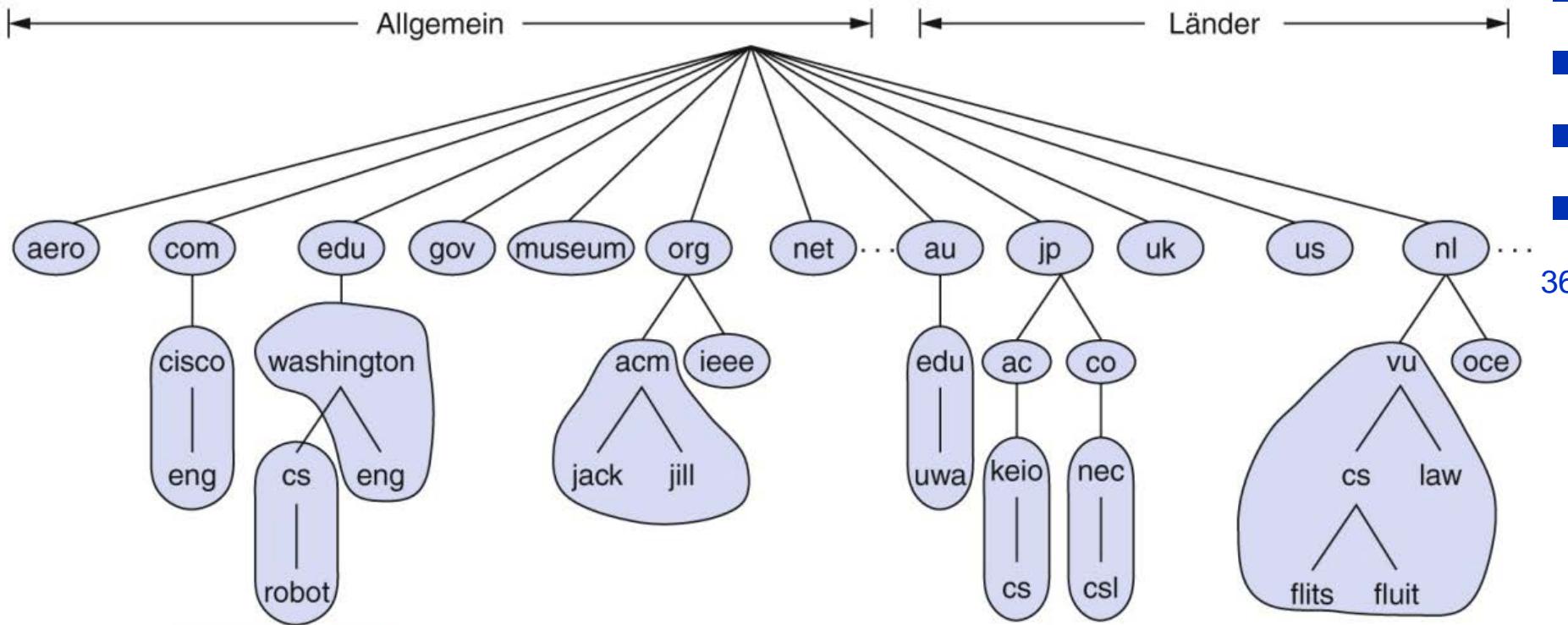
- Hierarchisch unterteilt in Domänen und Teildomänen



35

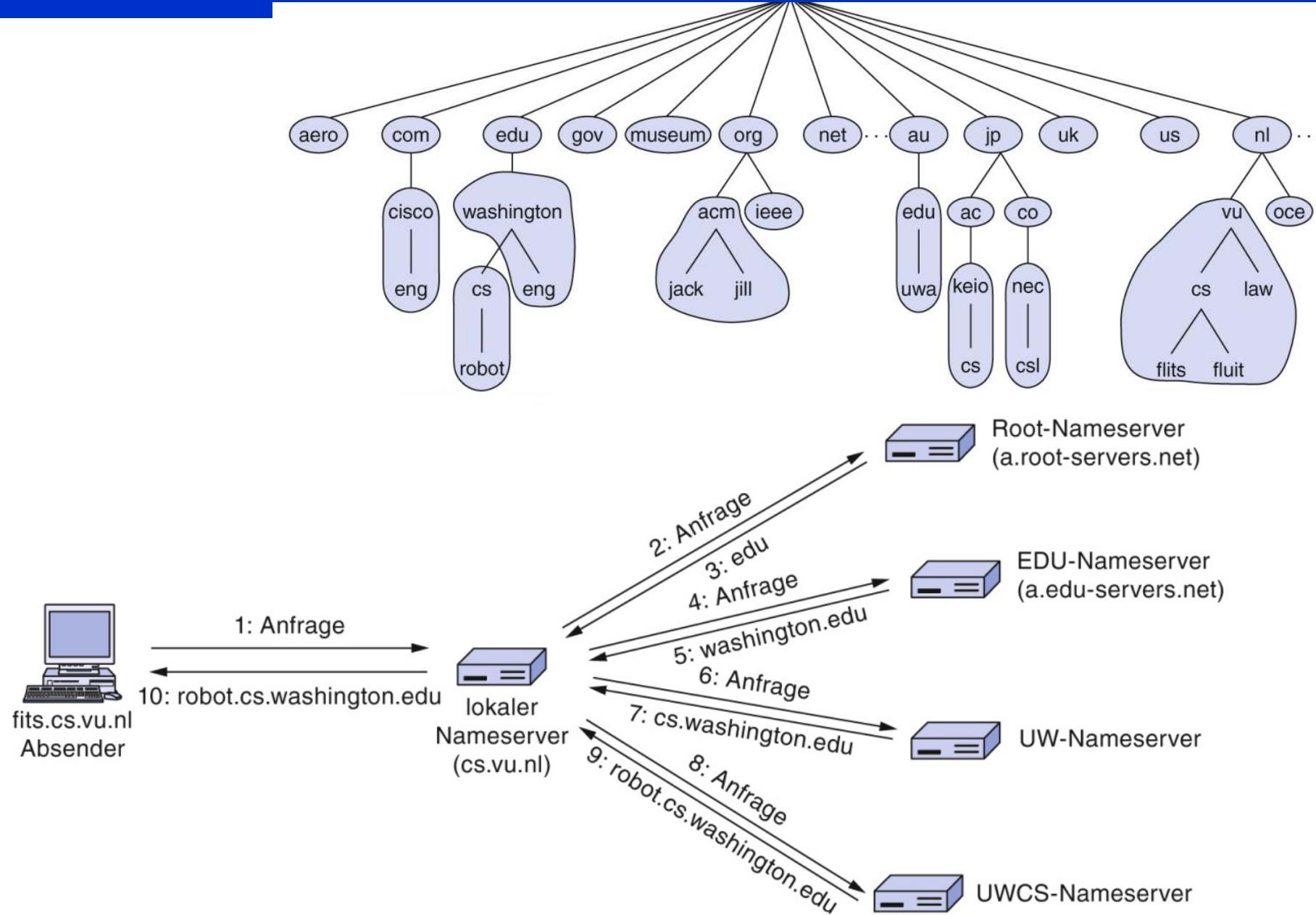
# ANWENDUNGSSCHICHT - DNS

- Aufteilung der Namensbereiche in Zonen (im Bild eingekreist)



- In jeder Zone wird ein Name Server verwaltet, der die DNS-Namen seiner Zone in IP-Adressen auflösen kann und Name Server „benachbarter“ Zonen kennt

# DNS: BEISPIELANFRAGE



# ANWENDUNGSSCHICHT

- E-Mail / Simple Mail Transfer Protocol (SMTP)
  - E-Mail-Adressen:
  - meier@uni-bonn.de
- Web / Hypertext Transfer Protocol (HTTP)
  - Adressen/URLs:
  - <http://www.uni-bonn.de/index.htm>

38

# FRAGEN?

39

# IT-SICHERHEIT

## 2. GRUNDLEGENDES

Wintersemester 2015/16

Prof. Dr. Michael Meier  
Informatik IV  
Arbeitsgruppe IT-Sicherheit



Universität Bonn  
E-Mail: mm@cs.uni-bonn.de

# ÜBERBLICK

- Begriffe
- Sicherheit
- Interessen
- Evaluierungskriterien
- Angreifermodelle
- Ansätze für Sicherheitsmechanismen

2

- **IT-System:** technisches System mit der Fähigkeit zur Speicherung und Verarbeitung von Informationen
- **Information:** wird durch **Daten** repräsentiert und ergibt sich durch eine festgelegte Interpretation der Daten

5

- **Objekte:** repräsentieren Informationen
  - passive Objekte (z.B. Dateien): Fähigkeit zur Speicherung von Informationen
  - aktive Objekte (z.B. Prozesse): Fähigkeit zur Speicherung und Verarbeitung von Informationen
  - **Assets:** Informationen und Objekte, die sie repräsentieren sind die schützenswerten Güter (asset) eines Systems

# IT-SYSTEM (2)

- **Subjekte:** Benutzer oder aktive Objekte die im Auftrag von Benutzern aktiv sind (z.B. Prozesse, Server, Prozeduren)
- **Zugriffe:** Interaktionen zwischen einem Subjekt und einem Objekt durch die Informationsfluss auftritt
  - Zugriff auf Datenobjekt ist gleichzeitig Zugriff auf die dadurch repräsentierte Information

# SICHERHEIT UND VERLÄSSLICHKEIT

- Definitionen von Sicherheit
  - Funktionssicherheit (Safety): Ist-Funktionalität == Soll-Funktionalität
  - Informationssicherheit (Security): keine unautorisierte Informationsgewinnung oder Informationsveränderung
  - Datensicherheit (Protection): kein unautorisierter Zugriff auf Systemressourcen, insbesondere auf Daten
  - Datenschutz (Privacy): natürliche Person kontrolliert Erhebung und Verarbeitung ihrer persönlichen Daten (informationelle Selbstbestimmung)
- Definition: Verlässlichkeit (Dependability)
  - Funktionssicherheit **plus**
  - Funktion wird zuverlässig erbracht (Reliability)

# GRUNDLEGENDE SCHUTZZIELE

- Vertraulichkeit (Confidentiality)
  - keine unautorisierte Kenntnisnahme
- Integrität (Integrity)
  - keine unautorisierte unbemerkte Datenmanipulation
- Zurechenbarkeit (Accountability)
  - sicherheitsrelevante Aktivitäten eines Subjekts sind diesem eindeutig zurechenbar
- Verfügbarkeit (Availability)
  - autorisierte Subjekte können ihre Berechtigung ohne unautorisierte Beeinträchtigung wahrnehmen

# WEITERE BEGRIFFLICHKEITEN

- **Schwachstelle** (Weakness): Schwäche eines Systems, an dem es verwundbar ist
  - Diebstahlgefahr mobiler Geräte ist physische Schwachstelle
  - Naturkatastrophen sind natürliche Schwachstellen
  - Programme mit fehlender Eingabeüberprüfung sind softwarebedingte Schwachstellen
- **Verwundbarkeit** (Vulnerability): Schwachstelle, über die die Sicherheitsdienste umgangen, getäuscht oder unautorisiert modifiziert werden können
- **Exploit**: Vorgehensweise (z.B. Software oder Folge von Befehlen) zur Ausnutzung einer Verwundbarkeit
  - **Zero-Day-Exploit**: Exploit, der vor oder am gleichen Tag wie die Verwundbarkeit bekannt wird

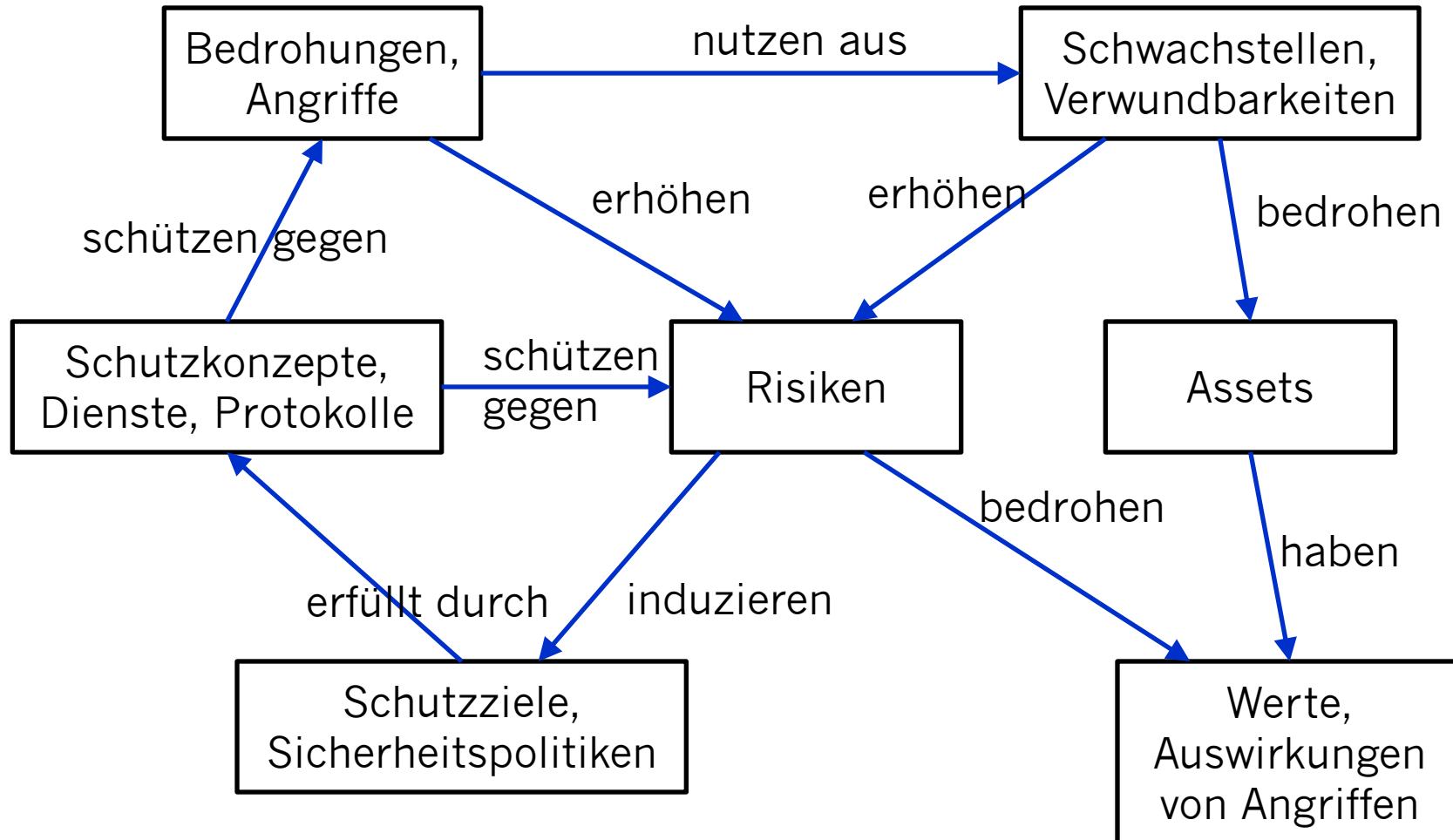
# WEITERE BEGRIFFLICHKEITEN

- **Bedrohung** (Threat): potentieller Verlust eines Schutzzieles durch Ausnutzung von Schwachstellen oder Exploits
- **Risiko** einer Bedrohung (Risk): Eintrittswahrscheinlichkeit (oder relative Häufigkeit) eines Schadensereignisses und Höhe des potentiellen Schadens
- **Angriff** (Attack, Intrusion)
  - nicht autorisierter Zugriff bzw. Zugriffsversuch

# SICHERHEITSPOLITIK

- Definition: **Sicherheitspolitik** (Security Policy):  
Menge von technischen und organisatorischen Regeln,  
Verhaltensrichtlinien, Verantwortlichkeiten, Rollen und  
Maßnahmen, um Schutzziele zu erreichen.

# ZUSAMMENHÄNGE



# SICHERHEITSMAßNAHME, -MECHANISMUS

- Definition: **Sicherheitsmaßnahme, -mechanismus** (safeguard, security measure):
  - Bestandteil der Systemarchitektur, der festgelegte Schutzziele durchsetzt und
  - die Verwaltung sicherheitsrelevanter Informationen und Konzepte realisiert

11

# AUFGASSIONEN VON SICHERHEIT

- Ein Computer-System ist sicher genau dann wenn es die vorgesehenen Zwecke erfüllt ohne relevante (informationelle oder andere) Rechte zu verletzen.
- Ein Computer-System gilt als sicher, wenn die verbleibenden Restrisiken tragbar sind.

12

# GRUNDLEGENDE SICHERHEITSINTERESSEN

- Verfügbarkeit von Daten und Aktivitäten
- Vertraulichkeit von Informationen und Aktionen
- Integrität von Computersystemen
- Authentizität von Akteuren
- Nicht-Abstreitbarkeit ihrer Aktion

13

# SICHERHEIT: ABGRENZUNG VON SECURITY & SAFETY

## Security

Schutz gegen beabsichtigte Angriffe

## Safety

Schutz vor unbeabsichtigten Ereignissen

### ■ Vertraulichkeit

- Abhörsicherheit
- Kein unbefugter Zugriff
- Anonymität
- Unbeobachtbarkeit

### ■ Integrität

- Übertragungsintegrität
- Zurechenbarkeit
- Abrechnungsintegrität

### ■ Verfügbarkeit

- Ermöglichen von Kommunikation

## Safety

Schutz vor unbeabsichtigten Ereignissen

Fehlertoleranz

### ■ Verfügbarkeit

- Funktionssicherheit
- Technische Sicherheit
- Schutz von Überspannung, Überschwemmung, Temperaturschwankungen
- Schutz vor Stromabfall

### ■ Sonstige Schutzziele

- Arbeitsschutz
- Maßnahmen gegen hohe Gesundheitsbelastung
- ...

# SCHUTZZIELE

- Klassische IT-Sicherheit berücksichtigt im Wesentlichen Risiken, die durch **regelwidriges Verhalten** in IT-Systemen entstehen

Vertraulichkeit

Unbefugter Informationsgewinn

Integrität

Unbefugte Modifikation

Verfügbarkeit

Unbefugte Beeinträchtigung der Funktionalität

15

# MEHRSEITIGE SICHERHEIT

- Mehrseitige Sicherheit bedeutet die Einbeziehung der **Schutzinteressen aller Beteiligten** sowie das Austragen daraus resultierender Schutzkonflikte.

Vertraulichkeit

Integrität

Verfügbarkeit

- Ziel:

- gegensätzliche Sicherheitsinteressen werden erkannt, Lösungen ausgehandelt und durchgesetzt

- Unter Umständen können nicht alle Interessen aller Beteiligten erfüllt werden

16

# SCHUTZZIELE DER MEHRSEITIGEN SICHERHEIT

Kommunikationsgegenstand  
Was? Worüber?  
Inhaltsdaten

Vertraulichkeit  
Verdecktheit

Inhalte

Integrität

Inhalte

Verfügbarkeit

Inhalte

Kommunikationsumstände  
Wann? Wo? Wer?  
Verkehrsdaten

Anonymität  
Unbeobachtbarkeit

Absender

Ort

Empfänger

Zurechenbarkeit  
Rechtsverbindlichkeit

Absender

Bezahlung

Empfänger

Erreichbarkeit

Nutzer

Rechner

# SCHUTZZIELE: DEFINITIONEN

- **Vertraulichkeit:** Geheimhaltung von Daten während der Übertragung. Niemand außer den Kommunikationspartnern kann den Inhalt der Kommunikation erkennen.
- **Verdecktheit:** Versteckte Übertragung von vertraulichen Daten. Niemand außer den Kommunikationspartnern kann die Existenz eines vertraulichen Inhalts erkennen.
- **Anonymität:** Nutzer können Ressourcen und Dienste benutzen, ohne ihre Identität zu offenbaren. Selbst der Kommunikationspartner erfährt nicht die Identität.
- **Unbeobachtbarkeit:** Nutzer können Ressourcen und Dienste benutzen, ohne dass andere dies beobachten können. Dritte können weder das Senden noch den Erhalt von Nachrichten beobachten.

# SCHUTZZIELE: DEFINITIONEN

- **Integrität:** Modifikationen der kommunizierten Inhalte (Absender eingeschlossen) werden durch den Empfänger erkannt.  
**Zurechenbarkeit:** Sendern bzw. Empfängern von Informationen kann das Senden bzw. der Empfang der Informationen bewiesen werden.  
(Wechselwirkungen zwischen Schutzz Zielen)
- **Verfügbarkeit:** Nutzbarkeit von Diensten und Ressourcen, wenn ein Teilnehmer sie benutzen will.  
**Erreichbarkeit:** Zu einer Ressource (Nutzer oder Maschine) kann Kontakt aufgenommen werden, wenn gewünscht.  
**Rechtsverbindlichkeit:** Ein Nutzer kann rechtlich belangt werden, um seine Verantwortlichkeiten innerhalb einer angemessenen Zeit zu erfüllen.

# EINSEITIGE ODER MEHRSEITIGE SICHERHEIT

Kommunikationspartner haben nicht immer gleiche Sicherheitsinteressen

## Kunde

Der Händler soll an meine Bestellung gebunden sein.

Ich möchte anonym bleiben, solange ich nichts kaufe.

Die Ware soll einwandfrei sein, sonst: Geld zurück!

Ich möchte anonym bleiben, beim Einkauf.

Händler soll keine Kunden-profile anlegen dürfen.

## Händler

Der Kunde soll an seine Bestellung gebunden sein

Der Kunde soll sich identifizieren.

Der Bezahlvorgang soll sicher sein (kein Betrug durch Kunden).

Digitale Signatur

Digitale Signatur

Vertrauen nötig

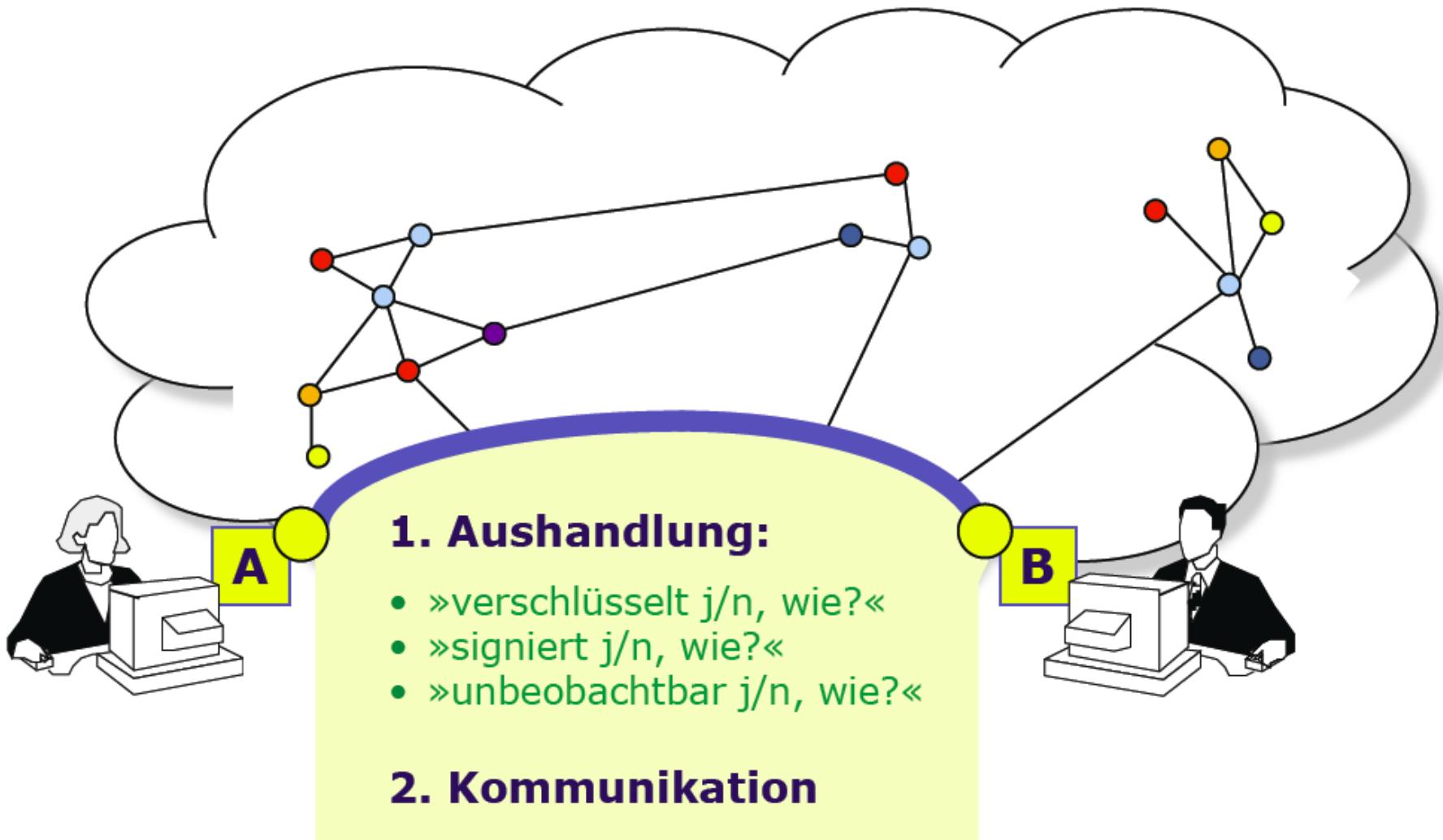
Pseudonymität:  
Treuhänder kennt Identität des Kunden, prüft Ware und Geld vor Lieferung

Anonyme Zahlungssysteme

Vertrauen nötig

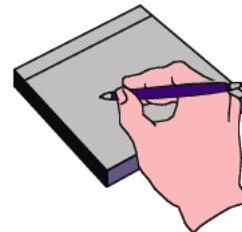
Selbstverpflichtung

20



# MEHRSEITIGE SICHERHEIT

- Definition: Mehrseitige Sicherheit bedeutet die Einbeziehung der Schutzinteressen aller Beteiligten sowie das Austragen daraus resultierender Schutzkonflikte beim Entstehen einer Kommunikationsverbindung.
- Vorgehen
  1. Sicherheitsinteressen formulieren
    - Setzt Verständnis der Benutzer voraus
    - Gute Bedienoberflächen erforderlich
  2. Konflikte erkennen und Lösungen aushandeln
    - Setzt entsprechende Tools und technische Protokolle voraus
  3. Sicherheitsinteressen durchsetzen
    - Beteiligte/Anwender brauchen Werkzeuge zum Selbstschutz
- Randbedingungen
  - möglichst wenig Vertrauen in andere setzen müssen,
  - Sicherheit mit minimalen Annahmen über andere



# GRUNDLEGENDE ASPEKTE VON SICHERHEIT

- Sicherheit ist umfassende Eigenschaft
- Sicherheitsentwurf reflektiert die **Interessen** von allen aktiven und passiven **Teilnehmern**
- Konflikte müssen balanciert werden
- Sicherheitsanforderungen eines Teilnehmers identifizieren vorgesehene **IT-Nutzung** und dabei vermutete **Bedrohungen**.

23

# GRUNDLEGENDE ASPEKTE VON SICHERHEIT

- Sicherheitsinteressen umfassen unter anderem
  - Verfügbarkeit von Daten und Aktivitäten
  - Vertraulichkeit von Informationen und Aktionen
  - Integrität von Computersystemen
  - Authentizität von Akteuren
  - Nicht-Abstreitbarkeit ihrer Aktionen
- Sicherheitsmechanismen zielen darauf
  - Sicherheitsverletzungen zu **vermeiden**
  - von Verletzungen verursachten Schaden zu **begrenzen**
  - die Konsequenzen von Verletzungen zu **kompensieren**

# GRUNDLEGENDE ASPEKTE VON SICHERHEIT

- Sicherheitsmechanismen sollten evaluiert werden
  - Ob oder in welchem Maße **erfüllen** Sicherheitsmechanismen die Sicherheitsanforderungen?
  - Welche **Annahmen** liegen der Evaluation zu Grunde?  
Welches **Vertrauen** wird Teilnehmern oder Systemkomponenten zugewiesen?
  - Rechtfertigen die erkannten Risiken die **Ausgaben/Aufwand** für die ausgewählten Sicherheitsmechanismen?

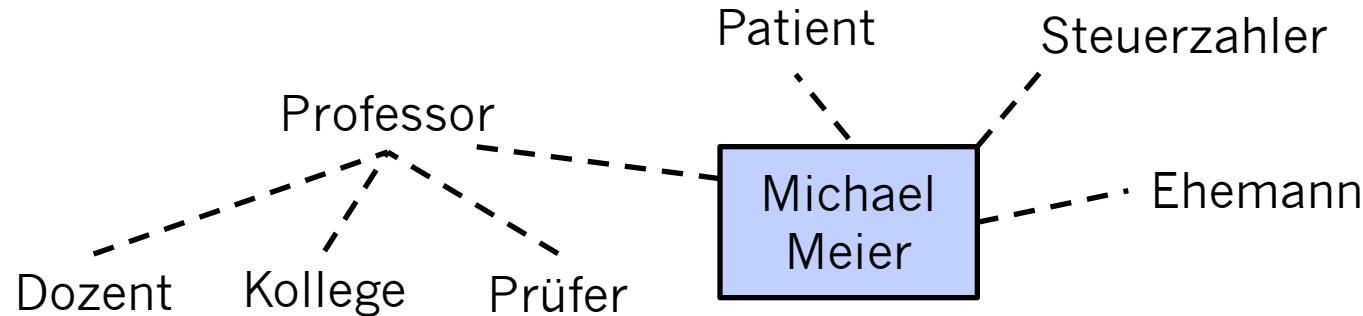
25

# ANFORDERUNGEN DER GESETZGEBUNG (BEISPIELE)

- Datenschutzgesetze konkretisieren die Prinzipien der **Informationellen Selbstbestimmung**
  - Verbot der Verarbeitung personenbezogener Daten mit Erlaubnisvorbehalt unter bestimmten Bedingungen
- Telekommunikations- und Dienste-Gesetze ermöglichen öffentliche und kommerzielle Nutzung und legen Grundlage für rechtsverbindliche Transaktionen
- Urheberrechtsgesetze übertragen Autoren/Urheber-Rechte auf digitale Medien
- Strafgesetzbuch identifiziert definitiv illegales Verhalten in Computersystemen und droht Strafe an

# DATENSCHUTZ UND INFORMATIONELLE SELBSTBESTIMMUNG

- Individuum bestimmt selbst welche persönlichen Informationen es bereit ist mit anderen zu teilen in einer spezifischen sozialen Rolle



- Individuum wählt seine sozialen Rollen eigenverantwortlich
- Andere respektieren die vorgesehene Trennung der Rollen und unterlassen unautorisierte Informationsflüsse zwischen verschiedenen Rollen

# SCHUTZZIELE DES DATENSCHUTZES

- Gegenstand des Datenschutzes: Schutz der einzelnen Personen vor Aktivitäten der Organisation
- Schutzziele des Datenschutzes
  - Transparenz
  - Nichtverkettbarkeit
  - Intervenierbarkeit

28

# SCHUTZZIELE DES DATENSCHUTZES

## ■ Transparenz

- Ziel: Verfahren mit Personenbezug können mit zumutbarem Aufwand nachvollzogen, überprüft und bewertet werden.
- Umsetzungsmaßnahme: die Komponenten eines Verfahrens vollständig dokumentieren und Abläufe protokollieren

29

# SCHUTZZIELE DES DATENSCHUTZES

- Nichtverkettbarkeit
  - Ziel: Daten mit Personenbezug können nicht oder nur mit unverhältnismäßig hohem Aufwand für einen anderen als den ausgewiesenen Zweck erhoben, verarbeitet und genutzt werden.
- Umsetzungsmaßnahme:  
Anonymisierungs- und Pseudonymisierungsverfahren.
  - Anonymisierung: das Verändern personenbezogener Daten derart, dass die Einzelangaben über persönliche oder sachliche Verhältnisse nicht mehr oder nur mit einem unverhältnismäßig großen Aufwand an Zeit, Kosten und Arbeitskraft einer bestimmten oder bestimmbaren natürlichen Person zugeordnet werden können.
  - Pseudonymisierung: das Ersetzen des Namens und anderer Identifikationsmerkmale durch ein Kennzeichen zu dem Zweck, die Bestimmung des Betroffenen auszuschließen oder wesentlich zu erschweren

30

# SCHUTZZIELE DES DATENSCHUTZES

- Intervenierbarkeit
  - Ziel: Betroffene können die ihnen zustehenden Rechte im Verfahren wirksam durchsetzen.
  - Umsetzungsmaßnahmen:
    - Betroffenen wird unmittelbar Zugriff auf ihre Daten und Prozesse gewährt.
    - Gesteuertes Changemanagement in Organisationen durch Integration von Datenschutzprozessen in Prozess-Frameworks und IT-Sicherheitsmanagement.

31

# SCHUTZREGELN FÜR PERSONENBEZOGENEN DATEN

- Erlaubniserfordernis
  - personenbezogene Daten sollen nur mit Erlaubnis verarbeitet werden
  - Erlaubnis kann durch Gesetz ausgedrückt werden oder durch explizite Zustimmung der betroffenen Person
- Datensparsamkeit (need-to-know)
  - Nur soviel wie nötig
  - Nur so lange wie nötig
- Zweckbindung
  - Verarbeitung nur für den Zweck, für den die Daten ursprünglich erhoben wurden
- Informationspflicht
  - Gegenüber betroffener Person, darüber welcher Art die Verarbeitung der Daten ist

32

# SICHERHEITSEVALUATIONS-KRITERIEN

- Von Sicherheitsbehörden herausgegebene Kriterienkataloge zur Unterstützung der kundenseitigen Bewertung der IT-Sicherheitsmerkmale von IT-Produkten
- **Common Criteria for Information Technology Security Evaluation**  
*(Allgemeine Kriterien für die Bewertung der Sicherheit von Informationstechnologie)*
  - Internationaler Standard
  - Unterscheidet zwischen **Funktionalität** eines Systems und dem Vertrauen (**Vertrauenswürdigkeit/Qualität**), das durch eine Prüfung in diese Funktionalität entstehen kann
  - Produkte werden von einer akkreditierten Prüfstelle evaluiert und bei einer Zertifizierungsstelle (in Deutschland Bundesamt für Sicherheit in der Informationstechnik) zertifiziert

33

# COMMON CRITERIA: FUNKTIONALITÄTSKLASSEN

- Bausteine zur Beschreibung der Sicherheitsfunktionalität
- (Security Functional Requirements – SFR)
  - FAU (Sicherheitsprotokollierung)
  - FCO (Kommunikation)
  - FCS (Kryptographische Unterstützung)
  - FDP (Schutz der Benutzerdaten)
  - FIA (Identifikation und Authentisierung)
  - FMT (Sicherheitsmanagement)
  - FPR (Privatsphäre)
  - FPT (Schutz der Sicherheitsfunktionen)
  - FRU (Betriebsmittelnutzung)
  - FTA (Schnittstelle)
  - FTP (vertrauenswürdiger Pfad/Kanal)

34

# COMMON CRITERIA EVALUATION ASSURANCE LEVEL (EAL) - VERTRAUENSWÜRDIGKEITSSTUFEN

- Charakterisieren die Korrektheit der Implementierung bzw. die Tiefe der betriebenen Überprüfungen
- EAL1: funktionell getestet
- EAL2: strukturell getestet
- EAL3: methodisch getestet und überprüft
- EAL4: methodisch entwickelt, getestet und durchgesehen
- EAL5: semiformal entworfen und getestet
- EAL6: semiformal verifizierter Entwurf und getestet
- EAL7: formal verifizierter Entwurf und getestet

35

# COMMON CRITERIA

- Zu Beginn einer Evaluation werden durch den Produkthersteller in einem Dokument (Security Target)
  - die Sicherheitsfunktionalität und
  - die zu prüfende Vertrauenswürdigkeit festgelegt.

36

# VOR WEM IST ZU SCHÜTZEN

- Angreifer
  - Außenstehende
  - Benutzer des Systems
  - Kommunikationspartner
  - Betreiber des Systems
  - Wartungsdienst
  - Produzenten des Systems
  - Entwerfer des Systems
  - Produzenten der Entwurfs- und Produktionshilfsmittel
  - Entwerfer der Entwurfs- und Produktionshilfsmittel
  - Produzenten der Entwurfs- und Produktionshilfsmittel der Produzenten und Entwerfer der Entwurfs- und Produktionshilfsmittel
  - Entwerfer der ...
  - ...



37

# ANGREIFERMODELL

- Schutz vor einem allmächtigen Angreifer ist unmöglich.

Das **Angreifermodell** definiert die maximal berücksichtigte Stärke eines Angreifers, gegen den ein Schutzmechanismus gerade noch wirkt

- Es beschreibt
  - **Rollen** des Angreifers (Außenstehender, Benutzer, Betreiber, Wartungsdienst, Produzent, Entwerfer ...), auch kombiniert
  - **Verbreitung** des Angreifers (Stellen im System, an denen der Angreifer Informationen gewinnen oder Systemzustände verändern kann)
  - **Verhalten** des Angreifers
    - passiv / aktiv, beobachtend / verändernd
  - **Rechenkapazität** des Angreifers
    - unbeschränkt: informationstheoretisch
    - Beschränkt: komplexitätstheoretisch

38

# VERHALTEN DES ANGREIFERS

	Aktiv	Passiv	Verletzbare Schutzziel
Beobachtend	(√)	√	Vertraulichkeit
Verändernd	√	∅	Vertraulichkeit, Integrität, Verfügbarkeit

39

# ANGRIFFSFORMEN

- Passive Angriffe
  - Lauschangriff (Eavesdropping)
  - Verkehrsflussanalyse (Traffic Analysis)
- Aktive Angriffe
  - Maskerade (Masquerading)
    - Man-in-the-middle attack
  - Verändern von Daten (Modification)
    - Einfügen von Daten (Injection, Spoofing)
      - • Wiederholen (replay)
      - • Fluten (Flooding, Spamming)
  - Dienstverweigerung (Denial of Service)

Vertraulichkeit

Integrität

Verfügbarkeit

# VERTRAUEN UND BEDROHUNGEN

- Teilnehmer sehen sich mitunter sowohl als gewünschten Partner als auch als **bedrohlichen** Gegner
- Mindestens etwas begrenztes **Vertrauen** muss einigen Teilnehmer entgegengebracht werden
- Komponenten eines Computersystems können Fehler machen, aber der Nutzer muss zumindest auf einige Komponenten **vertrauen**

41

# KRITISCHE PUNKTE DER MEHRSEITIGEN SICHERHEIT

- Das erforderlich Vertrauen sollte minimiert werden während gleichzeitig die erreichbare Funktionalität maximiert wird, wodurch potentielle Bedrohungen durch nicht vertrauenswürdige Teile besteht.
- Jeder Teilnehmer sollte autonom nach eigenem Ermessen Vertrauen zuweisen.
- Soweit möglich sollte zugewiesenes Vertrauen gerechtfertigt sein.
- Der zuweisende Teilnehmer sollte in der Lage sein die Vertrauenswürdigkeit zu verifizieren und das tatsächliche Verhalten des Vertrauensbereichs kontrollieren.

42

# ANNAHMEN - IN DER REALEN WELT

- Hinsichtlich der Absicherung eines Wohnhauses
  - Die Tür ist die einzige Zutrittsmöglichkeit. Es kann bspw. nicht durch die Fenster betreten werden.
  - Hersteller, Lieferanten und Verkäufer der Türen, Schlosser und Schlüssel agieren regelkonform und missbrauchen nicht das in sie gesetzte Vertrauen; keiner behält Duplikate der Schlüssel.
  - Die Hausbewohner verlieren keine Schlüssel.
  - Einbrecher werden durch Schutzmaßnahmen abgeschreckt oder scheitern daran, die Tür aufzubrechen.
- In der Praxis schwer zu erfüllen.



43

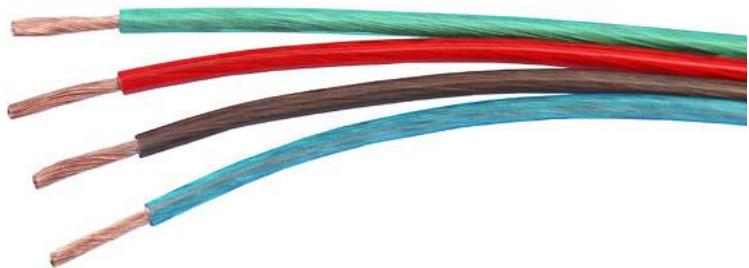
# ANNAHMEN - IN DER DIGITALEN WELT

- Die Sicherheitspolitik spezifiziert **exakt** die gewünschten erlaubten bzw. untersagten Zugriffe
- Die Verantwortlichen beschreiben die Sicherheitspolitik **korrekt** und **vollständig**.
- Die Sicherheitspolitik kann im IT-System **vollständig** repräsentiert werden.
- Die Kontrollsysteme können nicht umgangen werden und sie setzen die Sicherheitspolitik **ohne Ausnahme** durch.
- ...
- In der Praxis nur schwer und teilweise unmöglich zu erfüllen

# ANNAHMEN - IN DER DIGITALEN WELT

- Hard- und Software funktionieren fehlerfrei und enthalten keine Hintertüren.
  - 1000 Zeilen Programmcode enthalten durchschnittlich 3 Programmierfehler
    - Windows 2000: 29 Mio LoC; Mac OS X 10.4: 89 Mio LoC
- Intel, Microsoft, Apple und Co. agieren regelkonform und missbrauchen nicht das in sie gesetzte Vertrauen.
- Geheimnisse werden geeignet aufbewahrt.
- Biometrische Merkmale sind nicht kopierbar/simulierbar.
- Es gibt kein effizientes Faktorisierungsverfahren. (Bzw. es gibt keine Quantencomputer).

- **Redundanz** erlaubt
  - erlaubt die Ableitung erforderlicher Information
  - zur Erkennung von Fehlern und Angriffen sowie
  - zur Wiederherstellung nach unerwünschten Ereignissen.
- **Isolation** verhindert
  - unerwünschte Informationsflüsse.
- **Ununterscheidbarkeit**
  - lässt böswillige Beobachtungen zufällig oder standardisiert erscheinen und macht sie damit nutzlos.

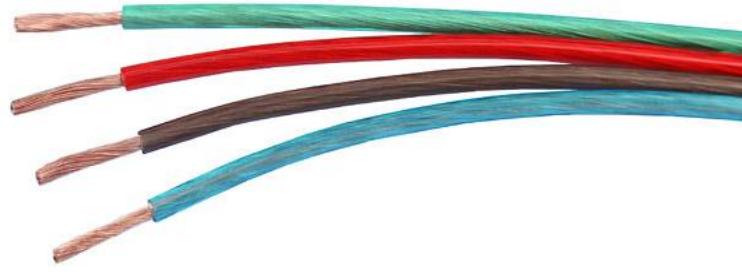


# BEISPIELE FÜR REDUNDANZ

- Reserve-Equipment und Notstrom
- Sicherheitskopien von Daten und Programmen
- Hinterlegung von Geheimnissen
- Netzwerke mit mehreren Verbindungen
- Fehler-erkennende und Fehler-korrigierende Kodierungen
- Diversität:
  - Redundante Komponenten verschiedener Herkunft
  - Unabhängige Entwicklung redundanten Komponenten

47

# ISOLATION



- Physikalische/Programmierungs-basierte Isolation
  - erfordert explizite Zugriffsentscheidungen zur Laufzeit
  - um beschränkte Nutzung der isolierten Komponenten entsprechend deklarierter Erlaubnisse/Rechte zu ermöglichen.
  - Klassische Zugangs- und Zugriffskontrolle, Firewalls, separate Prozessräume
- Virtuelle kryptographische Isolation
  - nutzt implizite Zugriffsentscheidungen
  - basierend auf der Verteilung von geheimen Schlüsseln.

48

# UNUNTERSCHEIDBARKEIT

- Verschleiert spezifische Aktivitäten, um sie ununterscheidbar von zufälligen oder standardisierten Ereignissen zu machen.
- Hindert unautorisierte Beobachter daran die Details oder gar das Auftreten spezifischer Aktivitäten abzuleiten.
- Kann erreicht werden durch Nutzung von
  - Zufall
    - Kryptographische Verschlüsselung: die Zufälligkeit des geheimen Schlüssels wird in die Zufälligkeit der zu schützenden Nachricht transformiert.
  - Standardisiertes Verhalten
    - Geeignetes standardisiertes Verhalten (z.B. Dummy-Aktivitäten, „Rauschen“) wird vorhersehbar generiert
    - Spezifische Aktivitäten werden im vorhersehbaren Verhalten verborgen (ersetzen z.B. einige der Dummy-Aktivitäten)

# KOMBINIERTE TECHNIKEN

## ■ Zugriffskontrolle und Monitoring

- Identifizierbare Akteure besitzen Zugriffsrechte
- Erlaubnisse und Verbote für Akteure definieren die **Sicherheitspolitik**
- Zugriffsanforderungen werden von Kontrollkomponenten abgefangen, die über Erlaubnis oder Ablehnung einer Zugriffsanforderung gemäß Sicherheitspolitik entscheiden.

## ■ Kryptographie

- Geheimnisse werden generiert und von Akteuren verwaltet
- Geheimnisse werden als kryptographische Schlüssel genutzt, die den **Schlüsselbesitzer** von anderen Akteuren unterscheiden, sodass dieser Akteur spezifische Operationen (sinnvoll) ausführen kann,  
im Gegensatz zu allen anderen Akteuren

# FRAGEN?

- Begriffe
- (Mehrseitige) Sicherheit
- Interessen
- Evaluierungskriterien
- Angreifermodelle
- Ansätze für Sicherheitsmechanismen

51

# IT-SICHERHEIT

## 3. PHYSISCHE SICHERHEIT

Wintersemester 2015/16

Prof. Dr. Michael Meier  
Informatik IV  
Arbeitsgruppe IT-Sicherheit

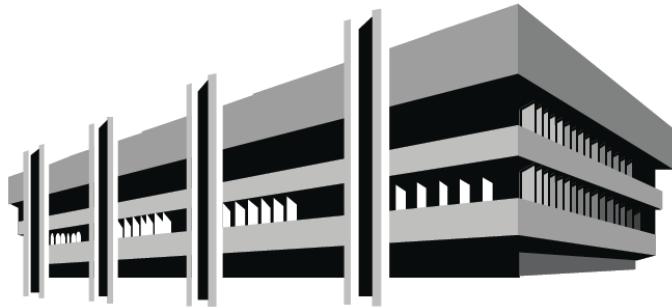


Universität Bonn  
E-Mail: mm@cs.uni-bonn.de

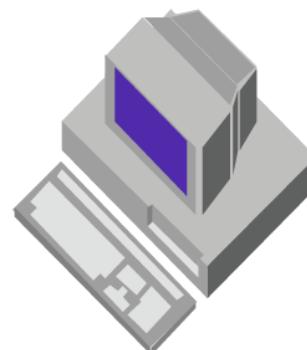
# PHYSISCHE SICHERHEIT

- Alle technischen Schutzmaßnahmen benötigen eine physische „Verankerung“ in Form einer Systemkomponente, auf die der Angreifer keinen physischen Zugriff hat.
- Physische Sicherheit zu erhalten, gelingt bestenfalls auf Zeit.
  - Verfügbarkeit nicht zu gewährleisten
    - ... nur ein Frage der Menge an Sprengstoff ...
  - Physischer Integritätsschutz möglich
    - erkennen von Manipulationen
  - Physischer Vertraulichkeitsschutz möglich
    - erkennen des unautorisierten Zugriffs und Selbstzerstörung der Information

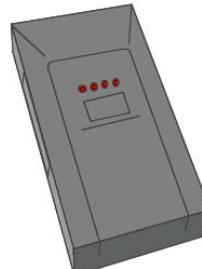
Rechenzentrum



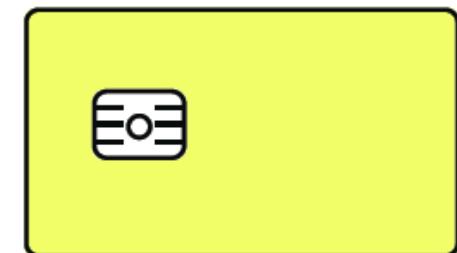
Einzelner  
Rechner



Sicherheitsmodul



Chipkarte



# PHYSISCHE SICHERHEIT EINZELNER RECHNER



3

# PHYSISCHE SICHERHEIT: GRUNDFUNKTIONEN

- Schutz gegen beobachtende Angriffe
  - Ausgaben
    - **Schirmung** (elektromagnetische Abstrahlung)
  - Eingaben
    - Energieverbrauch unabhängig von den zu schützenden Geheimnissen
- Schutz gegen verändernde Angriffe
  - **Erkennen, Bewerten, Verzögern** und ggf. **Löschen** der geheimen Information

Verzögern (z.B. hartes Material)  
Erkennen (z.B. Erschütterungs-, Drucksensoren)

Schirmen,  
Bewerten

Löschen

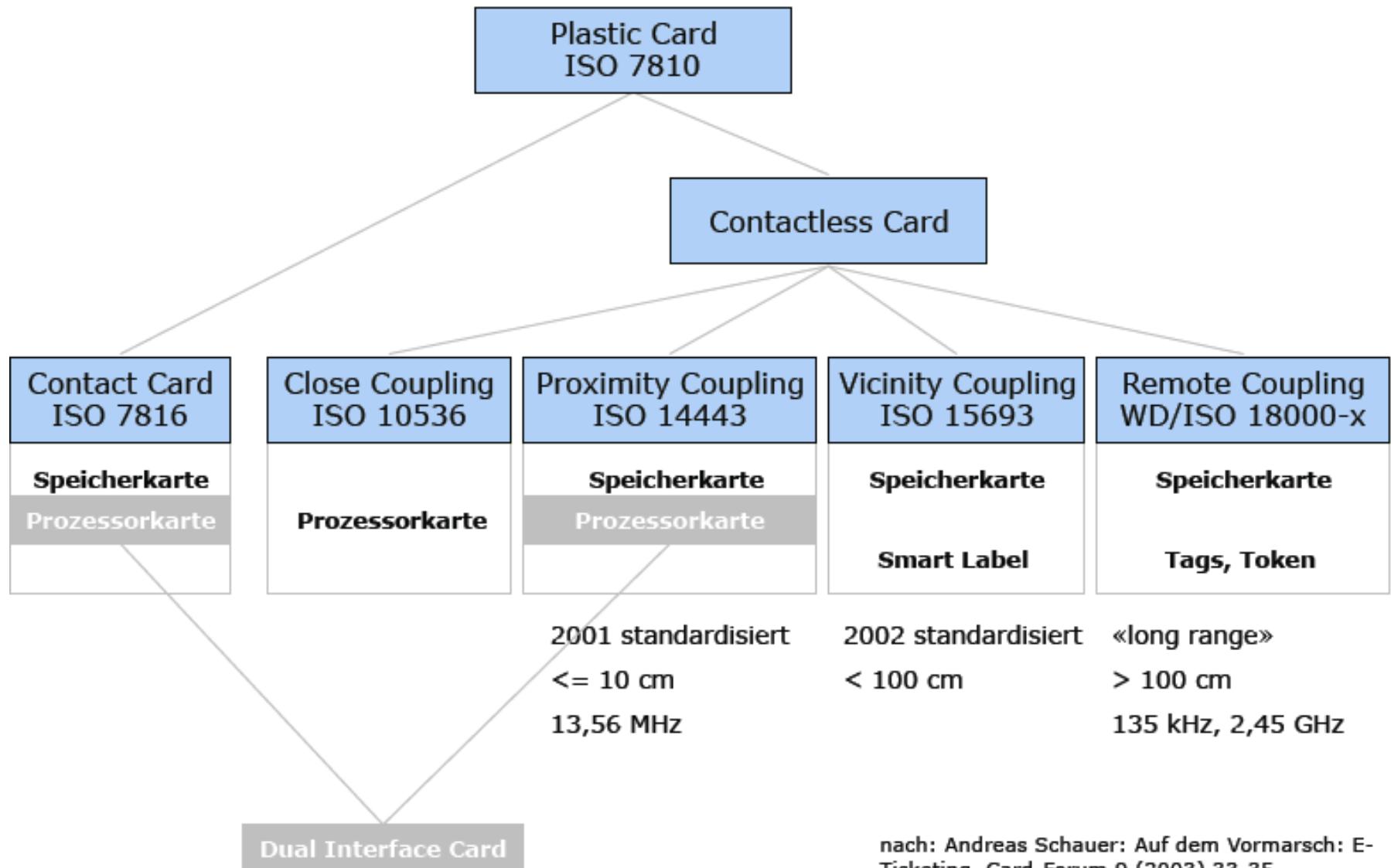
# PHYSISCHE SICHERHEIT:

## ■ Sicherheitsraum/modul



- Brandschutz
- Zutrittsschutz
- Klimatisierung
- Unabhängige Stromversorgung

# CHIPKARTEN



# (NEGATIV-)BEISPIEL CHIPKARTE

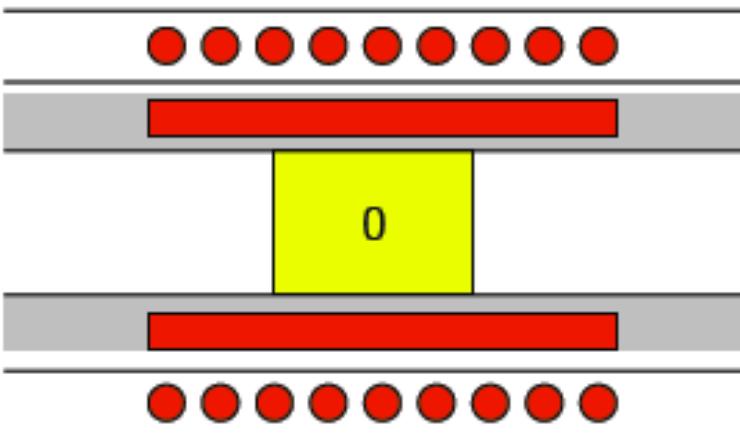
- Probleme
  - Erkennen schwierig
  - Schirmung schwierig (Karte dünn und biegsam)
  - Kein Löschen vorgesehen, selbst bei Stromversorgung
- Beispiele für Angriffe
  - Zerstörend
    - Vorbereitung: Abschleifen und Anätzen der Schutzschichten
    - Ggf. Reverse Engineering: Untersuchung der Strukturen unter Elektronenmikroskop, wenn Funktion unbekannt
    - Microprobing-Nadel
    - **Fault Injection:** gezielte Manipulation von Bits durch Beschuss mit elektromagnetischer Strahlung
  - Zerstörungsfrei: meist sog. **sidechannel attacks**
    - Messung des Energieverbrauchs: **power analysis**
    - Messung der benötigten Rechenzeit: **timing attacks**

7

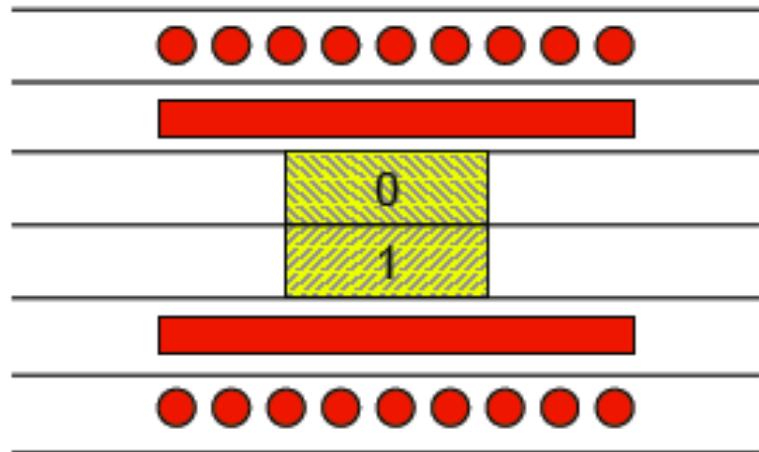
# ZERSTÖRENDE ANGRIFFE



Microprobing-Nadel



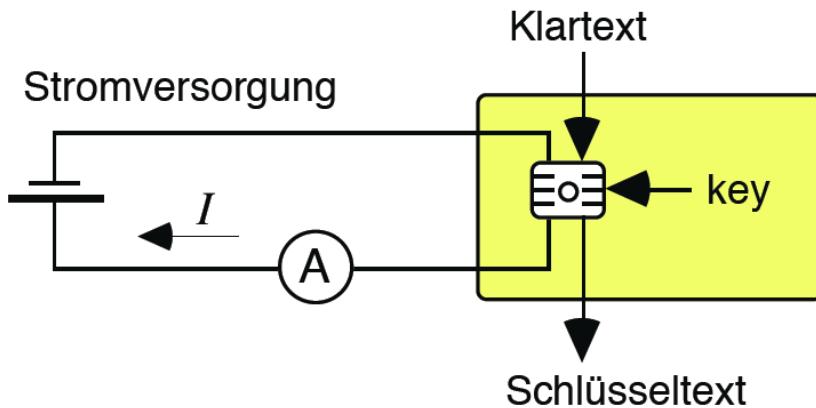
Fault Injection



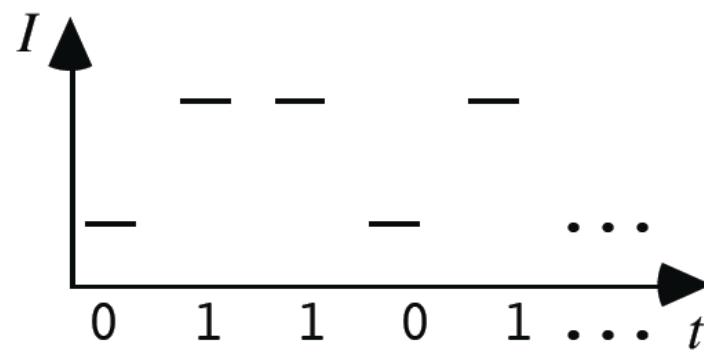
8

# ZERSTÖRFREIE ANGRIFFE

## TIMING ATTACK / POWER ANALYSIS (SKIZZE)



Angriffsziel: Key ermitteln



,1' hoher Stromverbrauch

,0' niedriger Stromverbrauch

Schlüsselbits sind direkt auslesbar

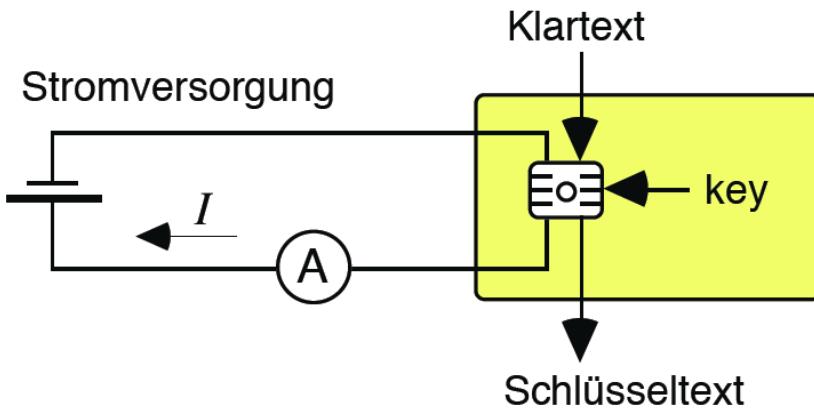
Dem Angreifer unbekannt:

```
key = array of bit  
= { 0, 1, 1, 0, 1, ... }
```

Dem Angreifer bekannt:

```
encrypt(Klartext, key)  
{ ...  
  for i = 1 to length(key)  
  { ...  
    if (key[i] == 1)  
    { ... }  
    /* ΔI > 0,  
     Δt > 0 */  
    else  
    { <nothing> }  
    /* I = const,  
     t = const */  
  }  
  ...  
}
```

# TIMING ATTACK / POWER ANALYSIS (SKIZZE)



Angriffsziel: Key ermitteln

Auswege: (Ununterscheidbarkeit)

- interner Energiepuffer
- Verzweigungen vermeiden
- <nothing> durch Dummy-Befehle mit exakt gleicher Zykluszeit wie { ... } ersetzen

Dem Angreifer unbekannt:

```
key = array of bit  
= { 0, 1, 1, 0, 1, ... }
```

Dem Angreifer bekannt:

```
encrypt(Klartext, key)  
{ ...  
  for i = 1 to length(key)  
  { ...  
    if (key[i] == 1)  
    { ... }  
    /* ΔI > 0,  
     Δt > 0 */  
    else  
    { <nothing> }  
    /* I = const,  
     t = const */  
  }  
  ...  
}
```

10

# COLD BOOT UND HOT PLUG: BRECHEN DER FESTPLATTENVERSCHLÜSSELUNG

- Cold Boot: <https://citp.princeton.edu/research/memory/>
  - Voraussetzung: Hauptspeicher des Rechners enthält den (flüchtigen) Key der Festplattenverschlüsselung
  - Nach Ausschalten verbleibt (nutzbare) Restladung
  - Abkühlen verlangsamt Speicherremanenz
  - Angreifer hat genügend Zeit, den Speicher auszubauen und auszulesen
- Hot Plug: <http://www1.informatik.uni-erlangen.de/sed>
  - Annahme: Festplattenverschlüsselung direkt auf der Platte  
    ⇒ Hauptspeicher enthält nicht mehr den Key
  - Hausdurchsuchungsszenario:  
    Rechner wird im laufenden Betrieb gefunden, nicht ausgeschaltet, die Festplatte (unter Strom) an einen externen Controller angeschlossen
  - einige Festplatten bemerken Controllerwechsel nicht

11

# IT-SICHERHEIT

## 4. AUTHENTIFIKATION

Wintersemester 2015/16

Prof. Dr. Michael Meier  
Informatik IV  
Arbeitsgruppe IT-Sicherheit



Universität Bonn  
E-Mail: mm@cs.uni-bonn.de

# IDENTIFIKATION UND AUTHENTIFIKATION

- Identifikation: Vorgeben einer Identität
- Authentifikation: Nachweisen der vorgegebenen Identität



# AUTHENTIFIKATION VON MENSCHEN DURCH IT-SYSTEME

## ■ Sprechweisen

- Das IT-System authentifiziert den Menschen: das IT-System überprüft die Identität des Menschen.
- Der Mensch authentifiziert sich gegenüber dem IT-System: der Mensch weist gegenüber dem IT-System seine Identität nach.

## ■ Ansätze

Authentifikation daran

- Was der Mensch ist
- Was der Mensch hat/besitzt
- Was der Mensch weiß

14

# AUTHENTIFIKATION VON MENSCHEN DURCH IT-SYSTEME

## ■ Was der Mensch ist

- Handgeometrie
- Fingerabdruck
- Aussehen (im Zshg. mit einem Ausweis)
- Eigenhändige Unterschrift (im Zshg. mit einem Ausweis)
- Retina-Muster
- Stimme
- Tipp-Charakteristik
- DNA-Muster

## ■ Was der Mensch hat/besitzt

- Papierdokument (im Zshg. mit einem Ausweis)
- Metallschlüssel
- Magnetstreifenkarte
- Chipkarte

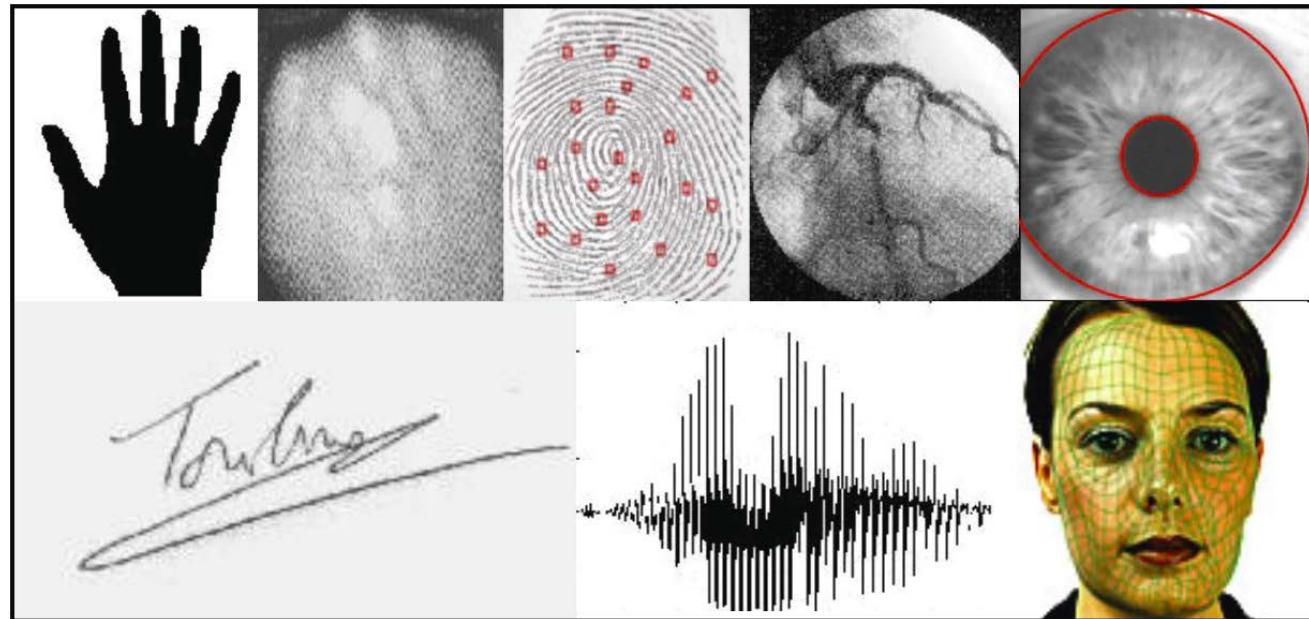
15

- Was der Mensch weiß
  - Passwort
  - Antworten auf Fragen
  - Rechenergebnisse für Zahlen
- Kombinationen verschiedener Ansätze sind sinnvoll.
- Wann wird authentifiziert?
  - Nur zu Beginn?
  - In regelmäßigen Zeitabständen?
  - Permanent?

# WAS DER MENSCH IST: BIOMETRISCHE MERKMALE

## ■ Physiologische

- Handgeometrie
- Handvenenmuster
- Fingerabdruck
- Retina
- Iris
- Gesicht
- DNA
- Ohrmuschel



## ■ Verhaltensbasierte

- Handschrift
- Stimme
- Lippenbewegung
- Tipp-Charakteristik
- Gang

# UMGEHUNG BIOMETRISCHER AUTHENTIFIKATION

The screenshot shows a web browser window with the following details:

- Title Bar:** BBC NEWS | Asia-Pacific | Malaysia car thieves steal finger - Windows
- Address Bar:** http://news.bbc.co.uk/2/hi/asia-pacific/4396831.stm
- Toolbar:** Includes icons for Back, Forward, Stop, Home, and a search bar containing "KeyGhost Keylogge...".
- Page Header:** BBC NEWS | Asia-Pacific | Malaysia car thieves steal finger - Windows
- Navigation Links:** BBC, Home, News (highlighted), Sport, Radio, TV, Weather, Languages.
- Version Selection:** UK version, International version, About the versions.
- Content Area:**
  - BBC NEWS logo:** A red globe icon.
  - One-Minute World News:** A video player with a play button and the text "WATCH One-Minute World News".
  - Last Updated:** Thursday, 31 March, 2005, 10:37 GMT 11:37 UK
  - Share Options:** E-mail this to a friend, Printable version.
  - Main Headline:** Malaysia car thieves steal finger
  - Byline:** By Jonathan Kent, BBC News, Kuala Lumpur
  - Summary:** Police in Malaysia are hunting for members of a violent gang who chopped off a car owner's finger to get round the vehicle's hi-tech security system.
  - Text Content:** The car, a Mercedes S-class, was protected by a fingerprint recognition system.  
Accountant K Kumaran's ordeal began when he was run down by four men in a small car as he was about to get into his Mercedes in a Kuala Lumpur suburb.

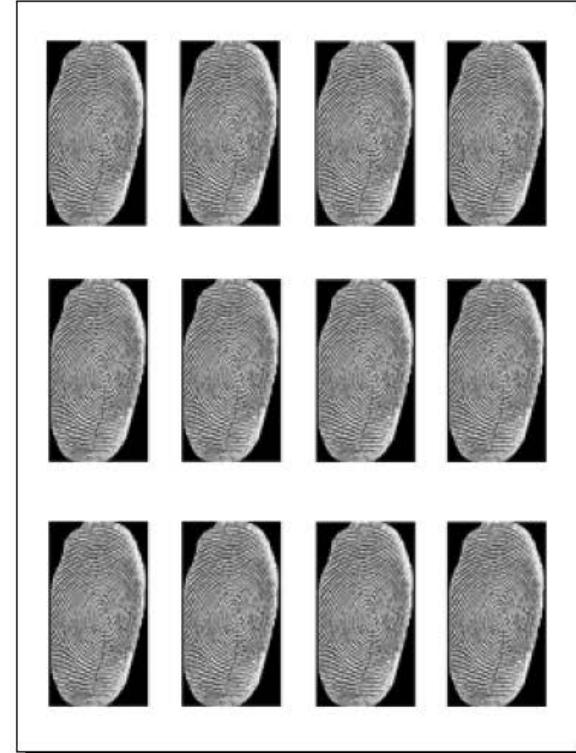
# UMGEHUNG BIOMETRISCHER AUTHENTIFIKATION



An Enhanced Fingerprint



A Fingerprint Image



A Mask with Fingerprint Images

*Yokohama Nat. Univ. Matsumoto Laboratory*

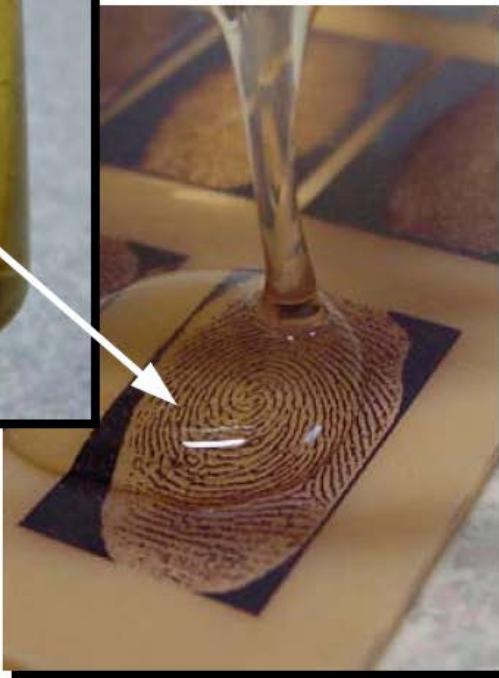
# UMGEHUNG BIOMETRISCHER AUTHENTIFIKATION

- (mold – Gussform)

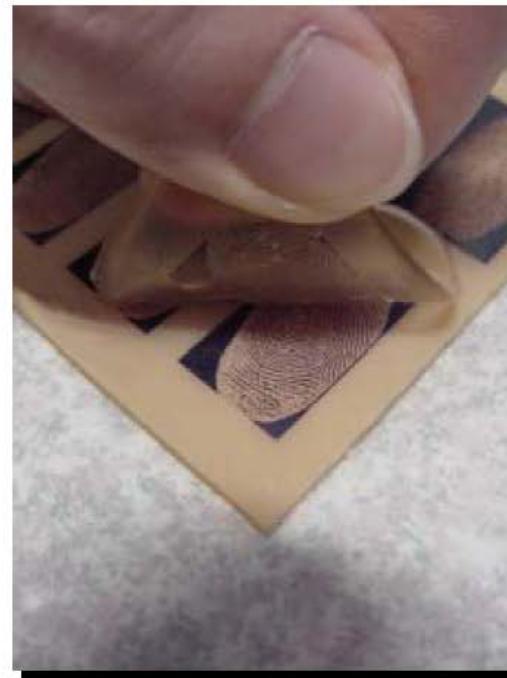
## Gelatin Liquid



Drip the liquid onto the mold.



Put this mold into  
a refrigerator to cool,  
and then peel carefully.



# UMGEHUNG BIOMETRISCHER AUTHENTIFIKATION

- Alternative zu Gelatine
  - ⇒ Knete
- Finger aus Knete überwinden 90% der Fingerprint Scanner (Clarkson University Studie)
  - ⇒ Schlagen Schweiß/Transpirationsmessung als Test für die Lebendigkeit des Fingers vor



21

# WAS DER MENSCH WEISS: PASSWORTREGELN

- Ändern Sie Ihr Passwort in regelmäßigen Abständen.
- Legen Sie niemals Passwörter (unverschlüsselt) in Dateien ab.
- Verwenden Sie in Ihrem Passwort nicht
  - Namen, Telefonnummern, Geburtsdaten, Autonummern
  - Wörter aus Wörterbüchern, Eigennamen
  - Tastaturmuster (vgl. „wertzuio“)
  - All dies rückwärts oder doppelt
  - All dies mit Ziffern oder Sonderzeichen davor oder dahinter
  - All dies in kombinierter Groß- und Kleinschreibweise
- Beachten Sie, dass häufig nur die ersten acht Zeichen des Passwortes signifikant sind
- Verwenden Sie
  - viele verschiedene Zeichen
  - kombinierte Groß- und Kleinschreibweise
  - Ziffern und Sonderzeichen
- Trick: Verwenden Sie die Anfangsbuchstaben eines „verrückten“ Satzes, der auch Zahlen und Sonderzeichen enthält.

# WELCHE PASSWÖRTER WERDEN TATSÄCHLICH GENUTZT?

- Im Dezember 2009 wurden 32 Millionen Passwörter von einem „Hacker“ veröffentlicht

- Ein blinder Log-in-Versuch mit „123456“ führt in 0,9% der Fälle zum Erfolg.
  - Rund 50% der Passwörter können als „schwach“ bezeichnet werden:
    - Wörter aus Wörterbüchern
    - Tastaturmuster
- ⇒ Prüfung der Passwortqualität durch Systembetreiber ist zwingend erforderlich!

Rang	Passwort	Häufigkeit
1	123456	290731
2	12345	79078
3	123456789	76790
4	Password	61958
5	iloveyou	51622
6	princess	35231
7	rockyou	22588
8	1234567	21726
9	12345678	20553
10	abc123	17542
11	Nicole	17168
12	Daniel	16409

23

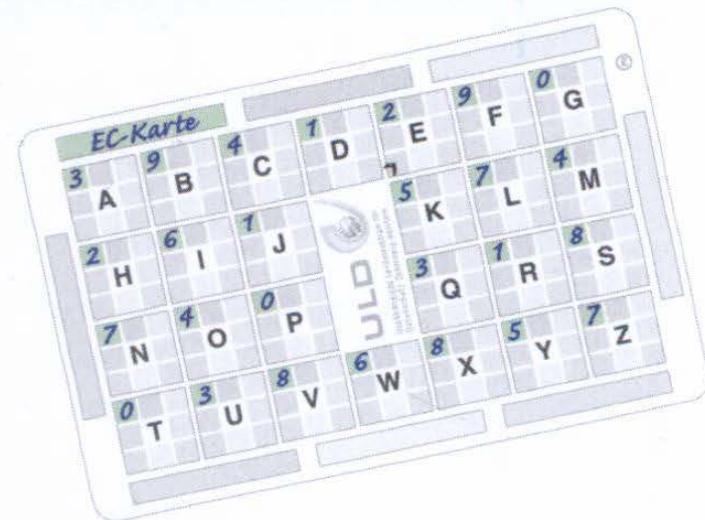
PINs	
EC-KARTE	2225
KREDITKARTE	3818
ONLINE-BANKING	323571
HANDY	0666
AUTORADIO	4275
TRESOR	842150
SCHLIESSFACH	74726
PAKET DEPOT	45217

Die clevere Alternative zum Merkzettel:  
Verschlüsseln Sie bis zu acht  
PINs und Passwörter.

**ULD**



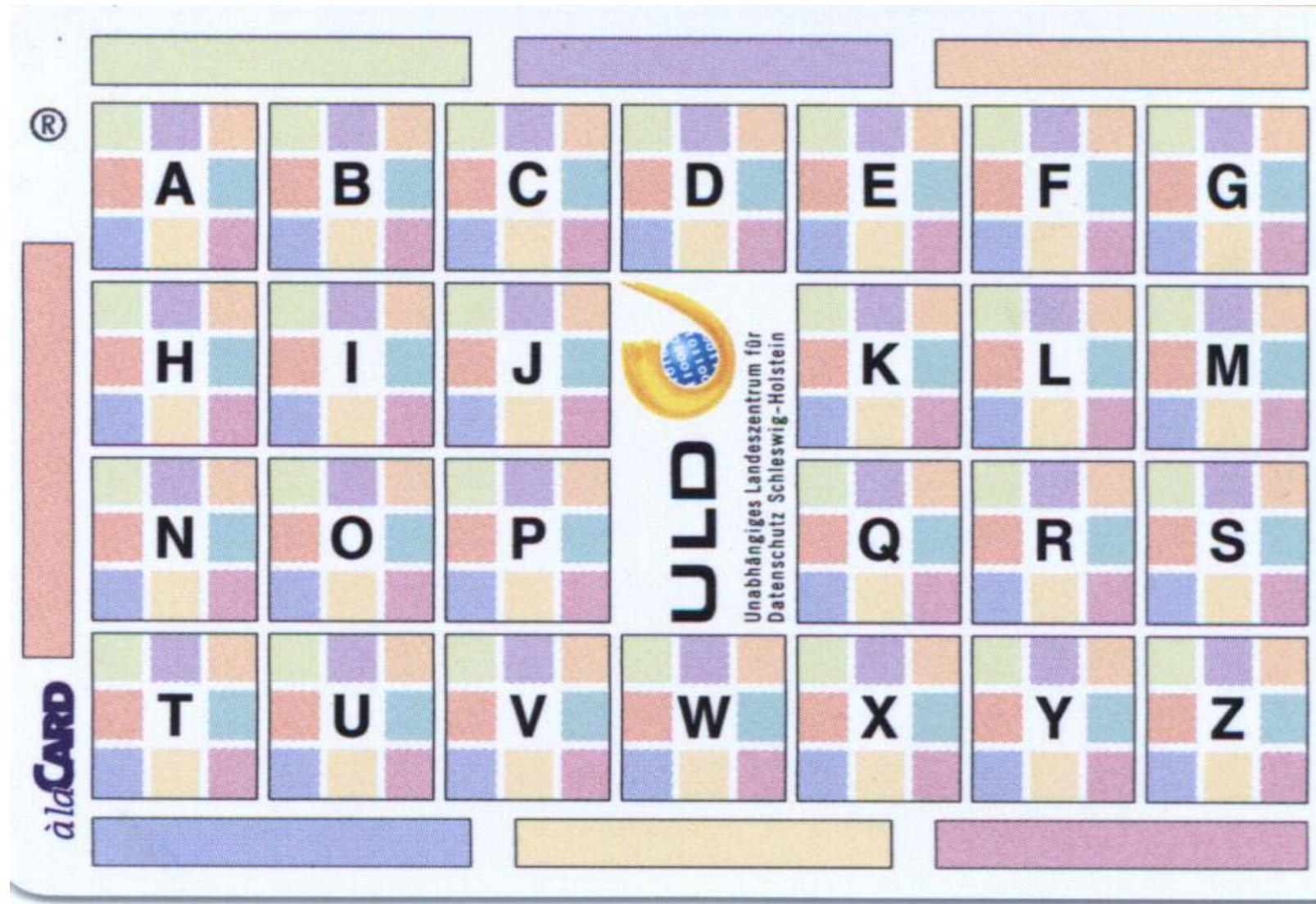
Unabhängiges Landeszentrum für  
Datenschutz Schleswig-Holstein



**Der Passwortsafe  
für die Hosentasche**

[www.datenschutzzentrum.de](http://www.datenschutzzentrum.de)

# PASSWORTSAFE FÜR DIE HOSENTASCHE

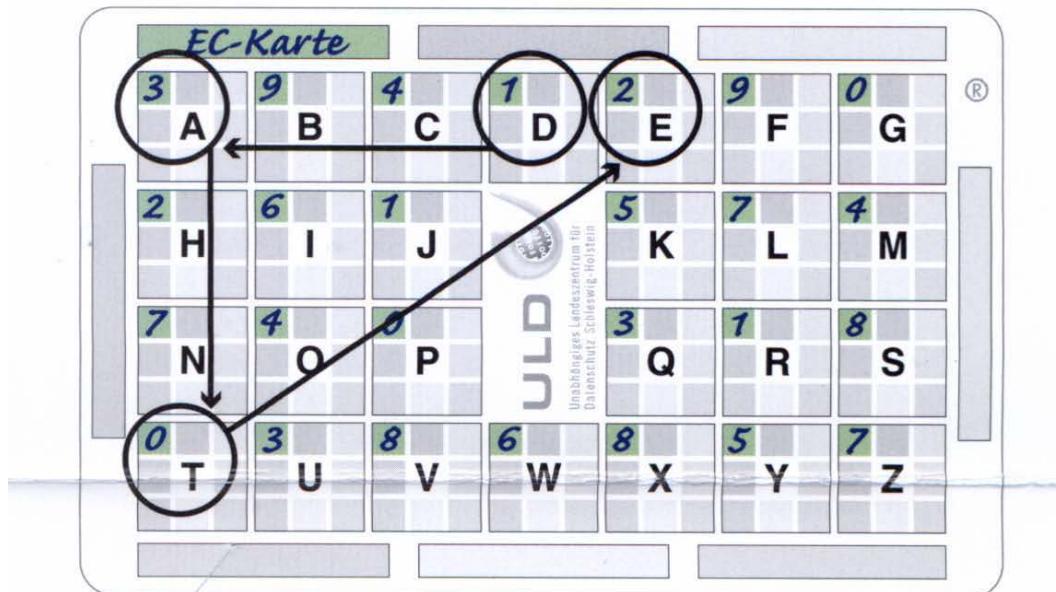


IT  
SEC



## Anwendungsbeispiel *äldeCARD*

Herr Meier hat die Geheimzahl seiner EC-Karte 1-3-0-2 unter dem ausgedachten Codewort »Datenschutz« in den grünen Farbfeldern versteckt. Bei den Buchstaben D-A-T-E findet er seine Geheimzahl wieder (grün = oben links).

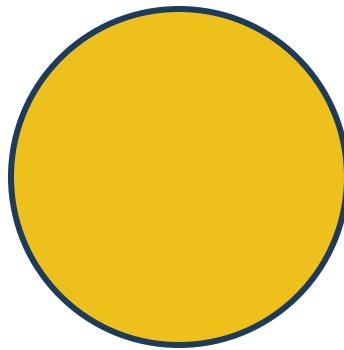


Die restlichen Buchstaben von »Datenschutz« wurden nicht benötigt. Zum Verstecken hat er alle anderen grünen Felder mit Zahlen von 0 bis 9 ausgefüllt. Für weitere 7 Geheimzahlen sucht er sich nur eine andere Farbe aus. Das Wort »Datenschutz« bleibt.

Tipps: Ihr Codewort sollte so ausgewählt sein, dass es auch im engsten Freundeskreis nicht erraten werden kann. Buchstabenwiederholungen innerhalb des Codewortes werden einfach übersprungen. Sie können statt eines Wortes auch ein optisches Muster verwenden. Wir empfehlen, zum Beschriften einen wasserfesten Faserstift zu verwenden.

# COGNOMETRICS

- Hirn verarbeitet Gesichter anders als andere Bilder



Gesichtserkennung ist ein spezifischer Prozess, der sich von allgemeiner Objekterkennung unterscheidet

# ERINNERN VS. ERKENNEN

Erinnern an ein Passwort



Erkennen eines Gesichts



Wie in der Schule ....

Füllen Sie den Freiraum

1 2 3 g f w y

---

*Welche Art Test bevorzugen Sie?*

Multiple Choice



# PASSFACES - ANSATZ

Gewöhnung des Nutzers an eine zufällig gewählte Menge von Gesichtern und prüfen ob er sich an diese erinnert

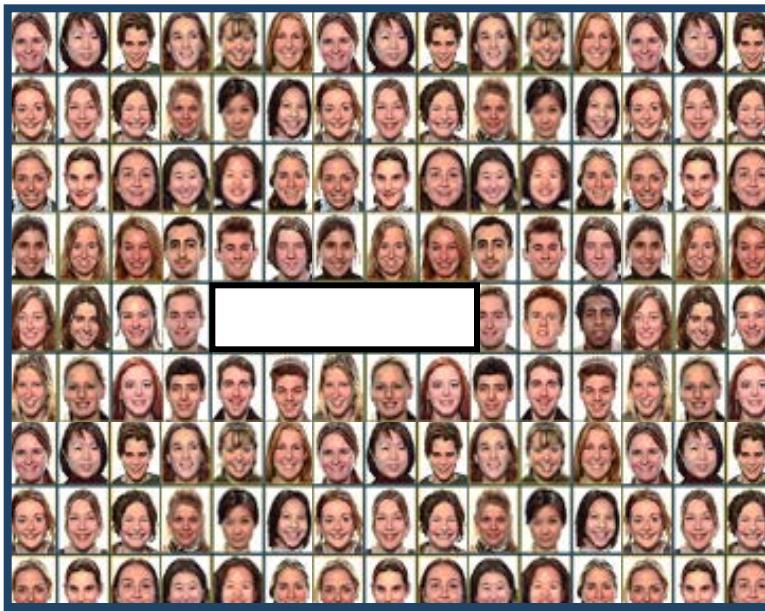
29



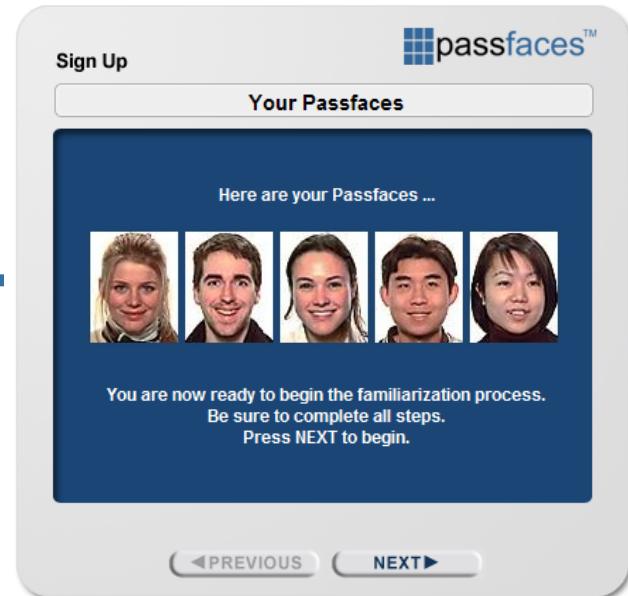
*It's as easy as recognizing an old friend*

# PASSFACES – WIE FUNKTIONIERTS

## Bibliothek von Gesichtern



## User Interface



Benutzer wird eine Menge mit 5 Passfaces zugeordnet

# PASSFACES – WIE FUNKTIONIERTS

- 5 Passfaces werden mit 40 “Ködern” dargestellt
- Passfaces werden angezeigt in fünf 3 x 3 Matrizen – jede mit einem Passface und 8 “Ködern”

31



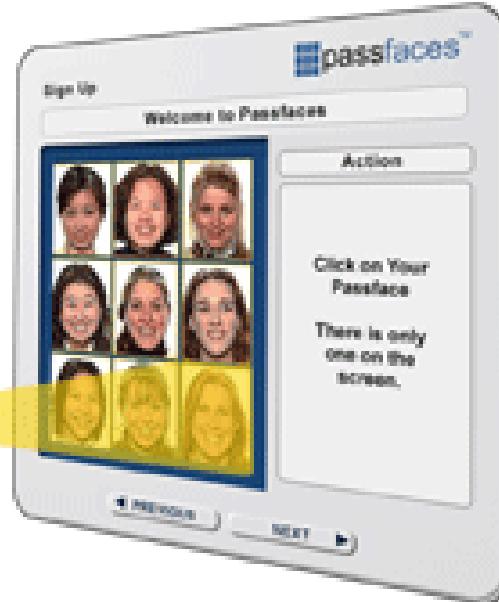
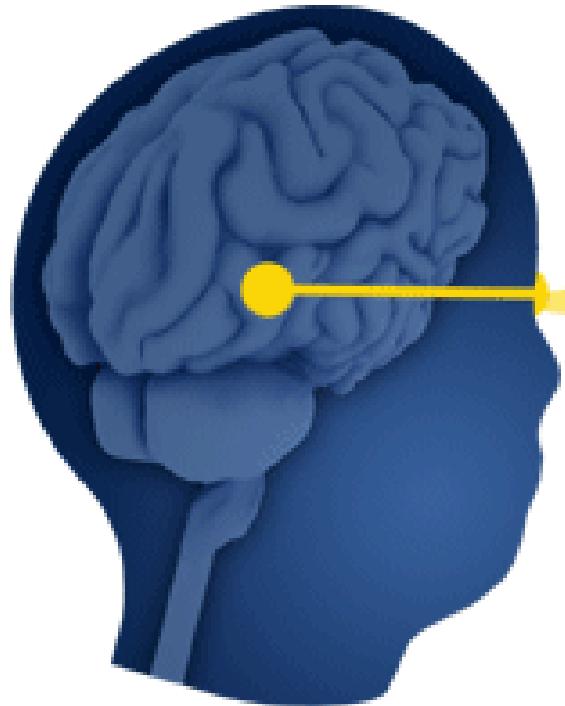
# PASSFACES – EINGEWÖHNUNG NEUER NUTZER



- Nutzer registrieren sich mit einem 2 bis 4 Minuten langen Gewöhnungsprozess
- Mittels direktem Feedback, Ermutigung, einfachen Dialogen werden Nutzer trainiert bis sie leicht ihre Passfaces erkennen
- Der Prozess ist optimiert und wird präsentiert wie ein einfaches Spiel

# EINE NEUE KLASSE DER AUTHENTIFIKATION

- Passfaces repräsentieren eine neue, vierte Klasse der Authentifikation  
Cognometrics („Kognometrie“)  
Erkennungs-Basierte Authentifikation



# EMPIRISCHE ERGEBNISSE

- Experimentelle Studie mit 154 Informatikstudierenden
- Schlussfolgerung
  - "... faces chosen by users are highly affected by the race of the user... the gender and attractiveness of the faces bias password choice... **In the case of male users, we found this bias so severe that we do not believe it possible to make this scheme secure against an online attack..."**
- 2 Versuche ausreichend für 10% der männlichen Benutzer
- 8 Versuche ausreichend für 25% der männlichen Benutzer

# NUTZERZITATE

- “I chose the images of the ladies which appealed the most”
- “I simply picked the best looking girl on each page”
- “In order to remember all the pictures for my login (**after forgetting my ‘password’ 4 times in a row**) I needed to pick pictures I could EASILY remember... So I chose beautiful women. The other option I would have chosen was handsome men, but the women are much more pleasing to look at”
- “I picked her because she was female and Asian and being female and Asian, I thought I could remember that”
- “I started by deciding to choose faces of people in my own race...”
- “... Plus he is African-American like me”

35

# AUTHENTIFIKATION VON IT-SYSTEMEN DURCH MENSCHEN

## ■ Ansätze

- Was das IT-System ist:
  - Gehäuse
  - Siegel
  - Hologramm
  - Verschmutzung
- Was das IT-System weiß:
  - Passwort
  - Antworten auf Fragen
  - Rechenergebnisse für Zahlen
- Wo das IT-System steht.



# MANIPULATION AN GELDAUTOMATEN

37



1. Aufsatz auf Kartenschlitz liest Magnetstreifen



# MANIPULATION AN GELDAUTOMATEN



2. Kamera erfasst PIN-Eingabe und überträgt die Daten per Funk

# PHISHING-EMAIL ES ANGREIFERS

The screenshot shows a web-based email client interface. The title bar reads "Für alle Volksbanken-Raiffeisenbanken Kunden — Eingang". The toolbar includes icons for deleting emails, marking as spam, reply, reply all, forward, and print. Below the toolbar, the email header information is displayed:

Von: VOLKSBANK <root@volksbank.de>  
Betreff: Für alle Volksbanken-Raiffeisenbanken Kunden  
Datum: 8. März 2007 11:53:17 MEZ  
An: federrath@sicherheit2005.de  
Antwort an: VOLKSBANK <root@volksbank.de>

The main body of the email contains the following text:

**Sehr geehrter Nutzer der Volksbanken Raiffeisenbanken Online-Bankings,**  
wir freuen uns Ihnen neue Informationen über die Sicherheit im Internet erteilen zu dürfen.  
Bitte lesen sie es aufmerksam!

Weltweit gilt das Online-Banking durch TAN Verfahren als eines der sichersten Legitimations-Verfahren für Online-Bankgeschäfte. Dennoch gab es in letzter Zeit immer wieder Versuche, auf betrügerische Art und Weise das Geld von Volksbanken Raiffeisenbanken Kunden ins Ausland zu überweisen.

Leider ist uns momentan das Verfahren, dass die Betruger benutzen, nicht bekannt.

Um unsere Kunden von Betrugern zu schützen, hat unser Sicherheitsteam für neue Schutzmaßnahmen entschieden.  
Beachten sie bitte, dass die Einsetzung dieser Schutzmaßnahmen erforderlich für alle Volksbanken Raiffeisenbanken Kunden ist!

Um diese Maßnahmen einführen zu können, müssen sie 2 TANs aus ihrer aktuellen Tan-Liste eingeben.

Folgen sie bitte diesen Link, um Ihr Konto bei der Volksbanken Raiffeisenbanken zu authentifizieren -  
<https://www.volksbank.vr-networld.de/frames/verify.php>

**Achtung! Wir bitten unsere Kunden um Verständnis für diese Überprüfung. Alle Volksbanken-Raiffeisenbankenkonten die nicht innerhalb eines Tages authentifiziert werden, werden gesperrt!**

39

# ORIGINAL

VR-NetWorld eBanking – Volksbank Möckmühl–Neuenstadt eG

<https://www.vr-networld-ebanking.de/index.php?RZKZ=X>

InternetBanking VR-NetWorld eBanking ...

 Volksbank Möckmühl-Neuenstadt eG

[zur Demo](#) | [AGB](#) | [Hilfe](#) | [Sicherheitshinweise](#)

### Anmeldung

Kundennummer  VR-NetKey

VR-NetKey:  oder Alias:

PIN:

Anmelden  Hilfe

**Hinweis:** Unsere Mitarbeiter werden Sie keinesfalls, weder per E-Mail noch telefonisch, dazu auffordern, Ihre Zugangsdaten in Verbindung mit Ihrer persönlichen PIN und/oder TAN preiszugeben.

Beachten Sie bitte unbedingt unsere Sicherheitshinweise.

 GÜNSTIGE TARIFE

# FÄLSCHUNG

InternetBanking

http://www.internetbankinggad.cd/volksbank-app/ Google

InternetBanking VR-NetWorld eBanking ...

Home Anmelden Demo Hilfe

**Anmeldung**

Bitte geben Sie Ihre Kontonummer sowie die zugehörige PIN ein

Bankleitzahl:

Kontonummer:

PIN:

Geben Sie bitte zwei früher nicht verwendete TAN ein.

TAN:

TAN:

Hilfe Weiter



**Hinweis:** Unsere Mitarbeiter werden Sie keinesfalls, weder telefonisch noch per eMail, dazu auffordern, Ihre Kontonummer in Verbindung mit Ihrer persönlichen PIN und/oder TAN preiszugeben.

Beachten Sie bitte unbedingt unsere Sicherheitshinweise.



# FÄLSCHUNG

InternetBanking

http://www.internetbankinggad.cd/volksbank-app/ Google

InternetBanking VR-NetWorld eBanking ...

Home Anmelden Demo Hilfe

**Anmeldung**

Vielen Dank! Ihr Online-Zugang wurde aktiviert!

 Browser  Sicherheit  VeriSign ON SITE

**Hinweis:** Unsere Mitarbeiter werden Sie keinesfalls, weder telefonisch noch per eMail, dazu auffordern, Ihre Kontonummer in Verbindung mit Ihrer persönlichen PIN und/oder TAN preiszugeben.

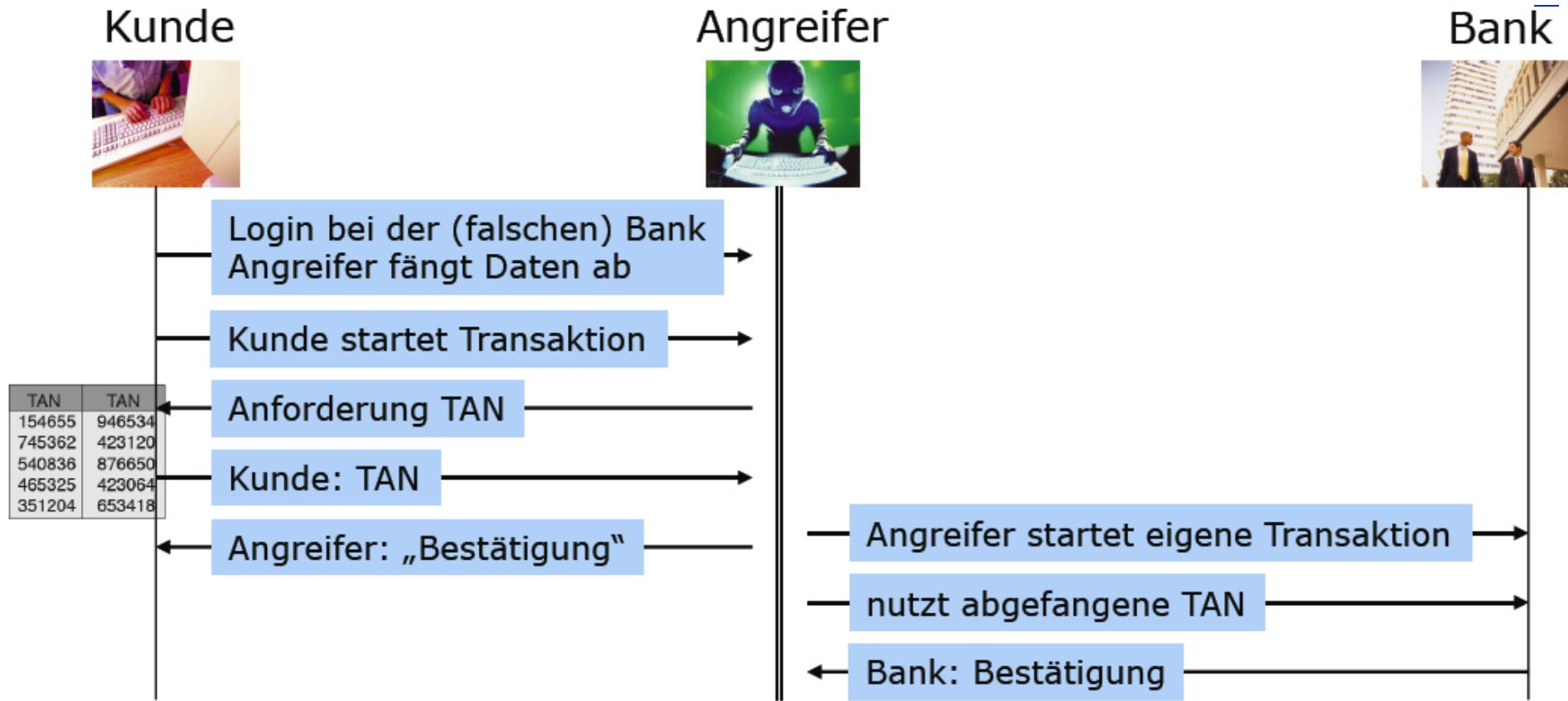
Beachten Sie bitte unbedingt unsere Sicherheitshinweise.

# MAN-IN-THE-MIDDLE-ATTACK AUF TAN-VERFAHREN (SKIZZE)

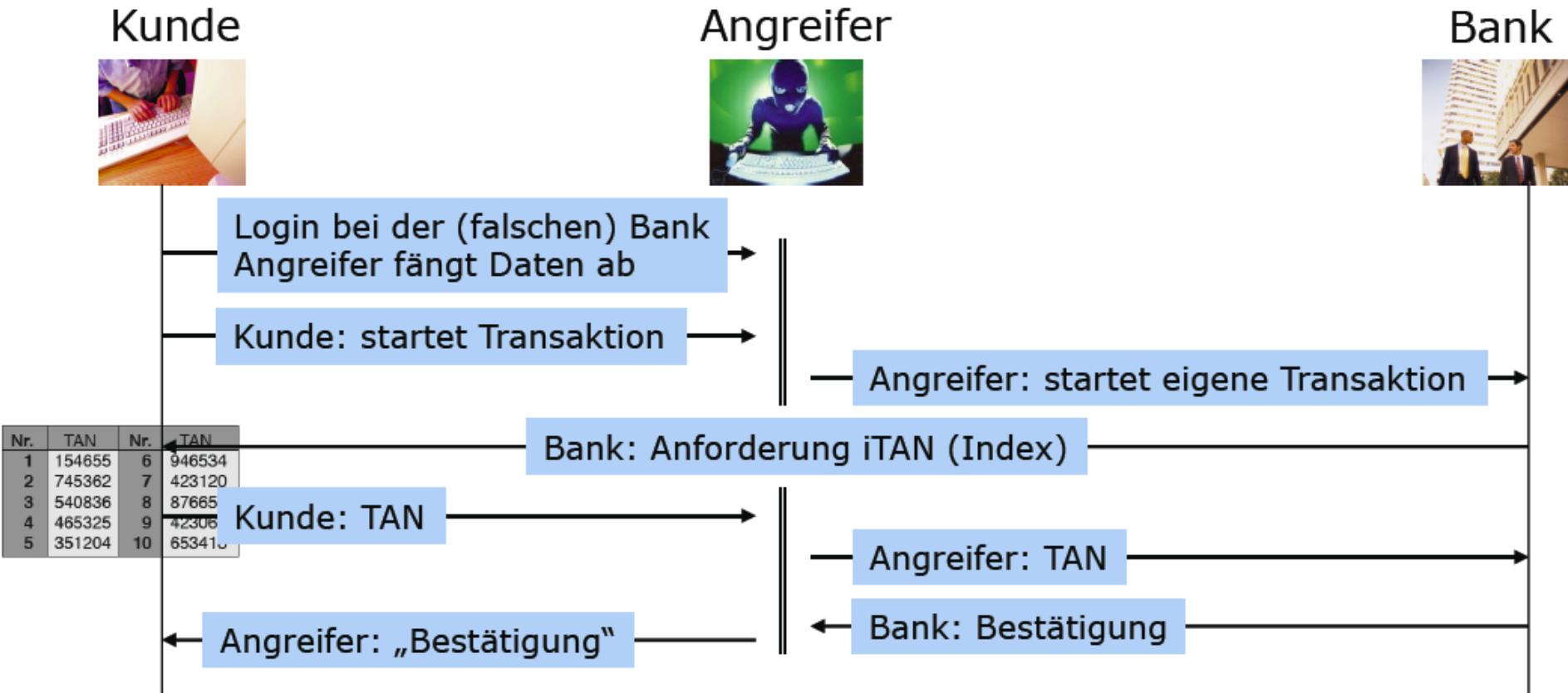
## ■ Voraussetzung: Angreifer

- betreibt täuschend echte Webseite der Bank
- bewegt den Kunden zum Besuch dieser Seite



# MAN-IN-THE-MIDDLE-ATTACKE AUF iTAN-VERFAHREN (SKIZZE)

- Verbesserung gegenüber normalem TAN-Verfahren:
  - Angreifer benötigt „Online-Hilfe durch Kunden“, d.h. er kann Transaktionen nur erfolgreich durchführen, wenn der Kunde dies selbst gerade tun will



# MTAN-VERFAHREN (SKIZZE)

## ■ Voraussetzung für Sicherheit:

- Mobiles Gerät wird nicht gleichzeitig für die Transaktion verwendet (Medienbruch beim Kunden)

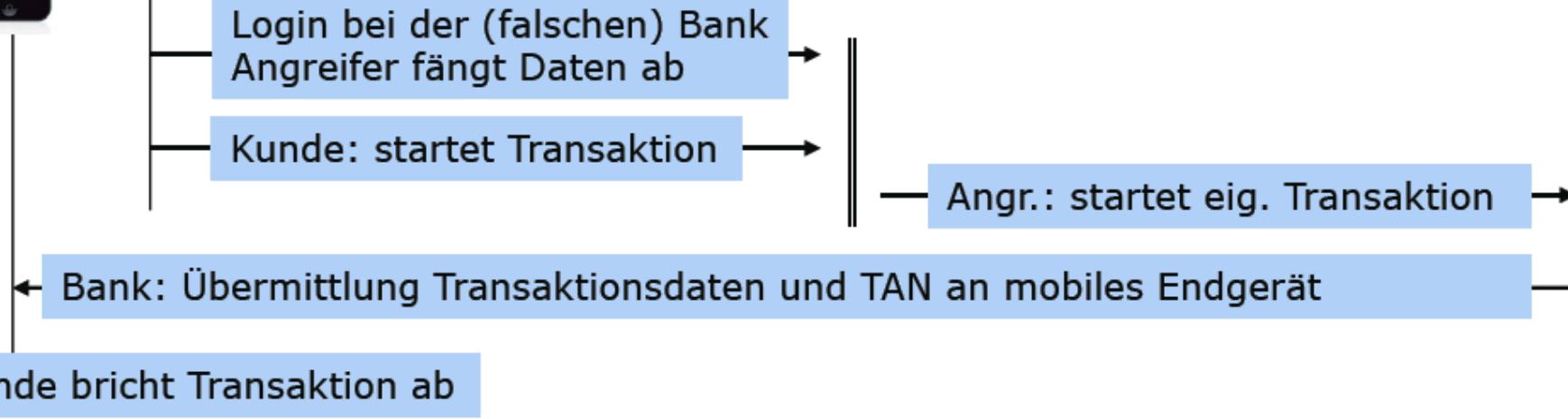
Kunde



Angreifer



Bank



# AUTHENTIFIKATION VON IT-SYSTEMEN DURCH IT-SYSTEME

- Was das IT-System weiß:
  - Passwort
  - Antworten auf Fragen
  - Rechenergebnisse für Zahlen
  - Kryptographie
- Leitung woher. (z.B. bei Chips auf dergleichen Platine)

46

# FRAGEN?

47

# IT-SICHERHEIT

## 5. ZUGRIFFSKONTROLLE

Wintersemester 2015/16

Prof. Dr. Michael Meier  
Informatik IV  
Arbeitsgruppe IT-Sicherheit



Universität Bonn  
E-Mail: mm@cs.uni-bonn.de

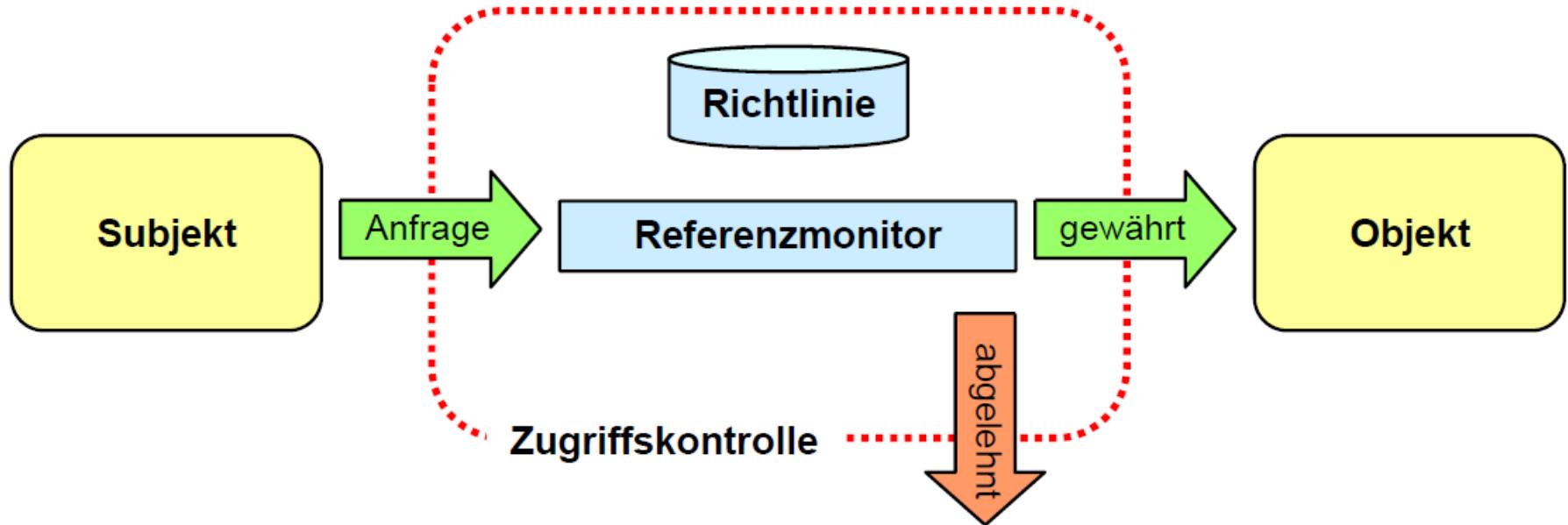
- Zugriffskontrollen
  - Benutzerbestimmbare Zugriffskontrolle (Discretionary Access Control)
    - Zugriffskontrollmatrix
    - Zugriffskontrolllisten (Access Control Lists)
    - Zugriffsausweise (Capabilities)
  - Systembestimmte Zugriffskontrolle (Mandatory Access Control)
    - Bell-La Padula Modell (Vertraulichkeits-Politik)
    - Biba Modell (Integritäts-Politik)
    - Chinese Wall Modell (Interessenkonflikte)

# WIEDERHOLUNG - BEGRIFFE

- **Objekte:** repräsentieren Informationen
  - passive Objekte (z.B. Dateien): Fähigkeit zur Speicherung von Informationen
  - aktive Objekte (z.B. Prozesse): Fähigkeit zur Speicherung und Verarbeitung von Informationen
- **Subjekte:** Benutzer oder aktive Objekte die im Auftrag von Benutzern aktiv sind (z.B. Prozesse, Server, Prozeduren)
- **Zugriffe:** Interaktionen zwischen einem Subjekt und einem Objekt durch die Informationsfluss auftritt
  - Zugriff auf Datenobjekt ist gleichzeitig Zugriff auf die dadurch repräsentierte Information

# ZUGRIFFSKONTROLLE

- Identifizierbaren Subjekten können Zugriffsrechte zugewiesen und entzogen werden



- Zugriffsanfragen von authentifizierten Subjekten werden vom Referenzmonitor
  - abgefangen
  - hinsichtlich deklarerter Zugriffsrechte/Richtlinien überprüft und
  - entsprechend gewährt oder abgelehnt

# DEKLARATION VON ZUGRIFFSRECHTEN

- **Zugriffsrecht:** Zugriffsmethode, die einem Subjekt auf ein Objekt gewährt werden kann
- Positiver Ansatz
  - nur explizite Erlaubnisse können formuliert werden
  - Standardfall: Verbot falls Erlaubnis abwesend (default-deny)
- Negativer Ansatz
  - nur explizite Verbote können formuliert werden
  - Standardfall: Erlaubnis falls Verbot abwesend (default-allow)
- Gemischter Ansatz
  - explizite Erlaubnisse und explizite Verbote können formuliert werden
  - Konfliktlösungsstrategie sind erforderlich

# ANFORDERUNGEN AN ZUGRIFFSKONTROLL-MECHANISMEN

- eindeutige, fälschungssichere Identifikation von Subjekten und Objekten
- keine unautorisierte Manipulation der Zugriffsrechte und Mechanismen
- vollständige Mediation aller Zugriffe (Unumgehbarkeit)
- Atomarität der Abfolge von Rechteprüfung und Zugriff
- Behandlung von Widersprüchen

6

# ZUGRIFFSKONTROLL-MODELLE

- Zugriffskontroll-Modell: legt die Ausdrucksfähigkeit für die Formulierung von Zugriffsbeschränkungen fest

Arten von Zugriffskontroll-Modellen

- **Benutzerbestimmbare Zugriffskontrollen**  
(Discretionary Access Control – DAC)
  - Eigentümerprinzip: Objekteigentümer entscheidet über Zugriffsrechte
  - Variante: **Rollenbasierte Zugriffskontrollen**  
(Role-based Access Control – RBAC): Zugriffsrechte an Rollen von Subjekten geknüpft
- **Systembestimmte Zugriffskontrolle**  
(Mandatory Access Control – MAC)
  - Systemregeln entscheiden über Zugriffrechte

# ZUGRIFFSKONTROLLMATRIZX

- Einfaches Modell zur Beschreibung von Zugriffsrechten
  - (Dynamische) Menge von Subjekten
  - (Dynamische) Menge von Objekten
  - Menge von Rechten

8

	Datei 1	Datei 2	Prozess 1	Prozess 2	Prozess 3	Prozess 4
Subjekt 1				control, send		control
Subjekt 2			wait, signal		control	
Subjekt 3	read, write	write, owner		receive		send
Subjekt 4		read, write			send	

# ZUGRIFFSKONTROLLMATRIZX

## ■ Eigenschaften

### Positiv

- sehr einfaches und intuitives Modell
- einfach zu implementieren

### Negativ

- Skaliert schlecht bei dynamischen Mengen von Subjekten, z.B. im Web-Umfeld
- Rechtevergabe ist an Subjekte, nicht an deren Aufgaben orientiert. Problem: Szenarien mit gleichen Subjekten, die über die Zeit wechselnde Aufgaben erfüllen, sind kaum modellierbar.
- Ungeeignet für Unternehmens- oder Verwaltungsumgebungen

# ACCESS CONTROL LIST (ACL)

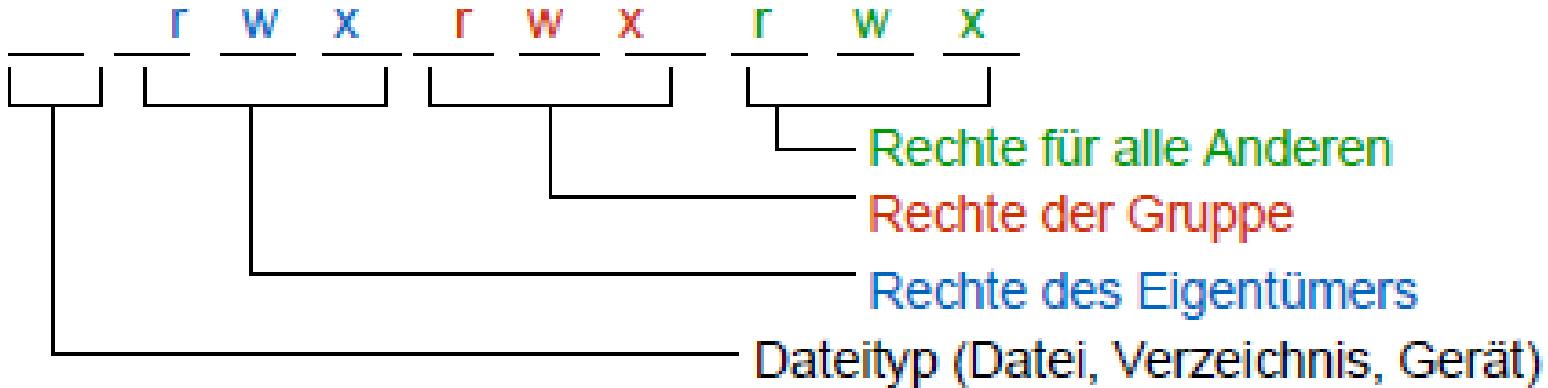
- Spaltenweise Realisierung der Zugriffskontrollmatrix
  - Objekten wird ACL zugeordnet, die Zugriffsrechte für angegebene Subjekte enthält
- Weit verbreitetes Konzept u.a. in Betriebssystemen
  - ACL ist eine geschützte Datenstruktur des Betriebssystems

10

	Datei 1	Datei 2	Prozess 1	Prozess 2	Prozess 3	Prozess 4
Subjekt 1				control, send		control
Subjekt 2			wait, signal		control	
Subjekt 3	read, write	write, owner		receive		send
Subjekt 4		read, write			send	

# ACCESS CONTROL LIST

- Beispiel ACL unter Unix/Linux:  
vereinfachte ACL  
Rechtevergabe nur an: Eigentümer, Gruppe, Rest der Welt



# VOR- UND NACHTEILE VON ACLS

## Positiv

- Einfache Verwendung

## Negativ

- Ineffizient bei langen Listen
- Fälschungssichere Speicherung auf Festspeicher erforderlich
  - Manipulation möglich
- Schlechte Skalierung bei dynamischer Menge von Subjekten (z.B. in verteilten Systemen)

12

# CAPABILITIES

- Zeilenweise Realisierung der Zugriffskontrollmatrix
  - Capability: Zugriffsticket mit Objekt-ID und Rechte-Bits
  - Capability-Besitz berechtigt zur Wahrnehmung der Rechte
  - Für jedes Subjekt wird eine Capability-Liste verwaltet

13

	Datei 1	Datei 2	Prozess 1	Prozess 2	Prozess 3	Prozess 4
Subjekt 1				control, send		control
Subjekt 2			wait, signal		control	
Subjekt 3	read, write	write, owner		receive		send
Subjekt 4		read, write			send	

# VOR- UND NACHTEILE VON CAPABILITIES

## Positiv

- Große Flexibilität
  - Zugriffskontrollprüfung nach Ausstellung einer Capability nur noch Prüfung der Zulässigkeit der Capability
- Dezentrale Verwaltung möglich
  - Trennung von
  - Komponenten zur Verwaltung der Information zur Capability-Ausstellung
  - Komponenten zur Prüfung der Zulässigkeit einer Capability
- Rechtedelegation möglich
  - Capability nicht an Subjekte gebunden
  - Können unter Einschränkungen an andere Subjekte weitergegeben werden (Zugriffs-Ticket)

## Negativ

- Rechteänderung nicht unmittelbar wirksam (siehe folgende Folien)

# KOMBINATION

- In heutige Systemen wir häufig eine Kombination aus beiden Ansätzen verwendet
  - ACL für den ersten Zugriff, danach
  - Ausstellen einer Capability:
    - File-Handle (Unix)
- Problem:
  - Rechteänderung nicht unmittelbar wirksam
  - Zugriffsrechteprüfung nur bei erstem Zugriff
  - Rechte können zwischenzeitlich entzogen worden sein

15

# ROLLENBASIERTE ZUGRIFFSKONTROLLE (RBAC)

- Ziel: effektive, skalierende, effiziente Rechteverwaltung
- Lösung: Aufgabenorientierte Rechtevergabe durch Rollen
- Rolle: beschreibt eine Aufgabe bzw. die damit verbundenen Verantwortlichkeiten, Pflichten und Berechtigungen
- Nachbilden von Organisationsstrukturen in Unternehmen
- Rechte und Verantwortlichkeiten sind häufig direkt aus den Organigrammen der HR Abteilungen ableitbar
- Entspricht den Prinzipien: need-to-know und separation-of-duty
- Weit verbreitet: u.a. Betriebssysteme, ERP, CMS, ...

16

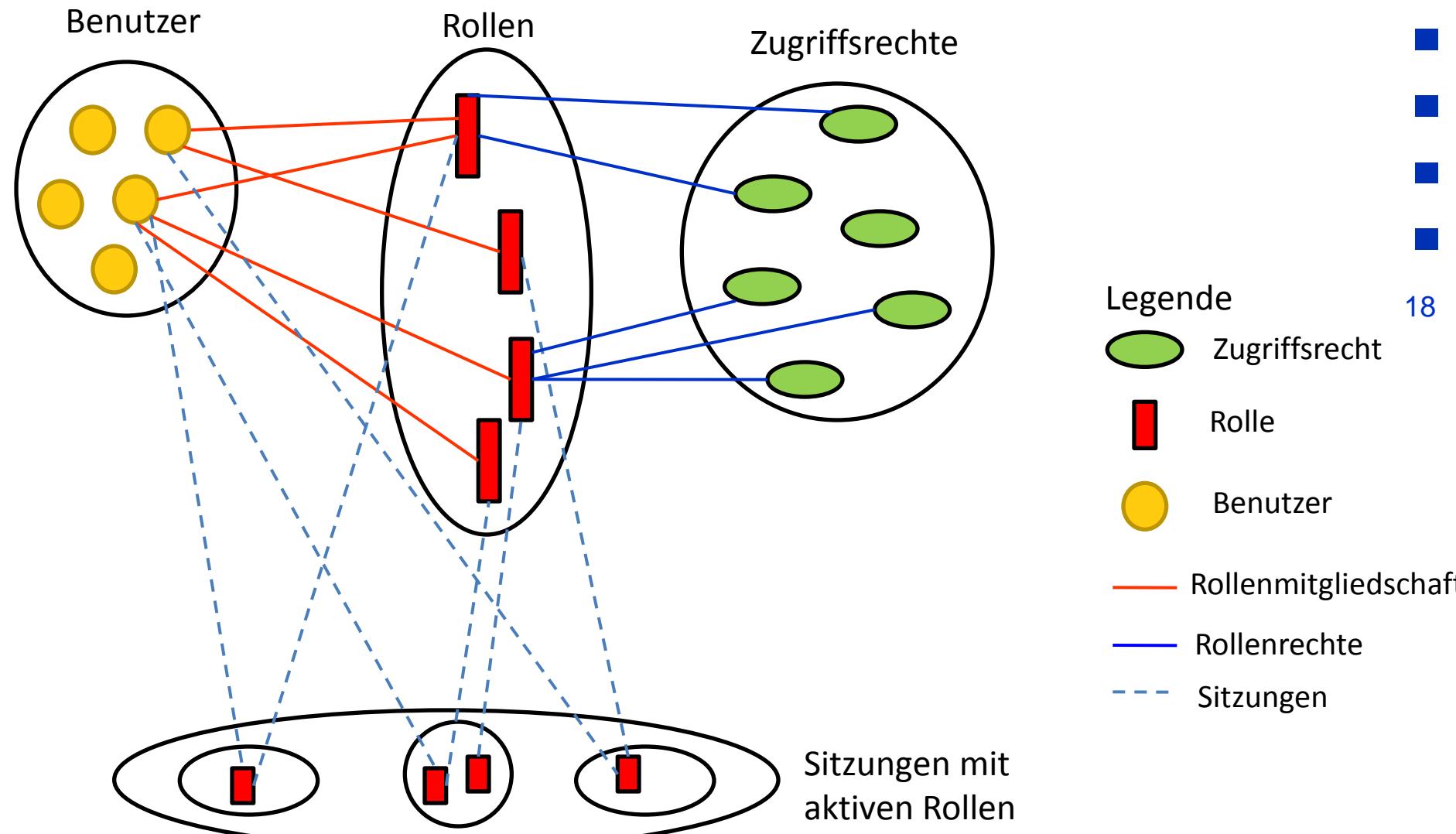
# KOMPONENTEN EINES (EINFACHEN) RBAC-MODELLS

- Subjekt
- Rollen
- Zugriffsrechte für Objekt
- Zwei Abbildungen
  - Zuordnung Subjekt – Rollen
  - Zuordnung Rollen – Zugriffsrechte
- Sitzung (= (Subjekt, Rollen)) charakterisiert einen Subjekt und dessen aktive Rollen

17

# ROLLENBASIERTE ZUGRIFFSKONTROLLE

18



# BEISPIEL: BANKSzenario

- Rollen: Zweigstellenleiter, Kassierer, Kundenbetreuer, Angestellter, Kassenprüfer, Kunde
- Subjekte
  - Klaus ist Zweigstellenleiter und Kunde
  - Petra ist Kundenbetreuer und Kunde
- Subjekt-Rollen-Zuordnung
  - Klaus: Zweigstellenleiter, Angestellter, Kunde
  - Petra: Kundenbetreuer, Angestellter, Kunde
- Rollen-Zugriffsrechte-Zuordnung
  - Zweigstellenleiter: Konto sperren, Kreditrahmen erhöhen
  - Kunde: Einzahlung auf Kundenkonto, Abheben von Kundenkonto
  - ...

# BENUTZERBESTIMMBARE ZUGRIFFSKONTROLLE

- Eigentümer eines Objekts kann beliebig Zugriffsrechte für andere Subjekte gewähren
- Problem: wie kann Weitergabe von Zugriffsrechten begrenzt werden
  - Gewährte Rechte können wieder anderen Subjekten gewährt werden
- Inhärente Schwäche Benutzerbestimmbarer Zugriffskontrolle
  - Unbeschränkte Benutzerbestimmbare Zugriffskontrolle erlaubt Informationsfluss von lesbaren Objekten in beliebige andere schreibbare Objekte
  - Vertrauen in Nutzer, dies nicht absichtlich zu tun, genügt nicht
  - Trojanisch Pferde können dennoch Information von einem Objekt zum anderen kopieren

20

# TROJANISCHES PFERD

- Programm, das eine vom Nutzer gewünschte Funktionalität besitzt, sowie eine vom Nutzer nicht gewünschte Funktionalität, die verborgen und ohne Einwilligung des Nutzers ausgeführt wird.



21

# TROJANISCHE PFERD: BEISPIEL

Nutzer A

Datei F

ACL

A:r  
A:w

Nutzer B

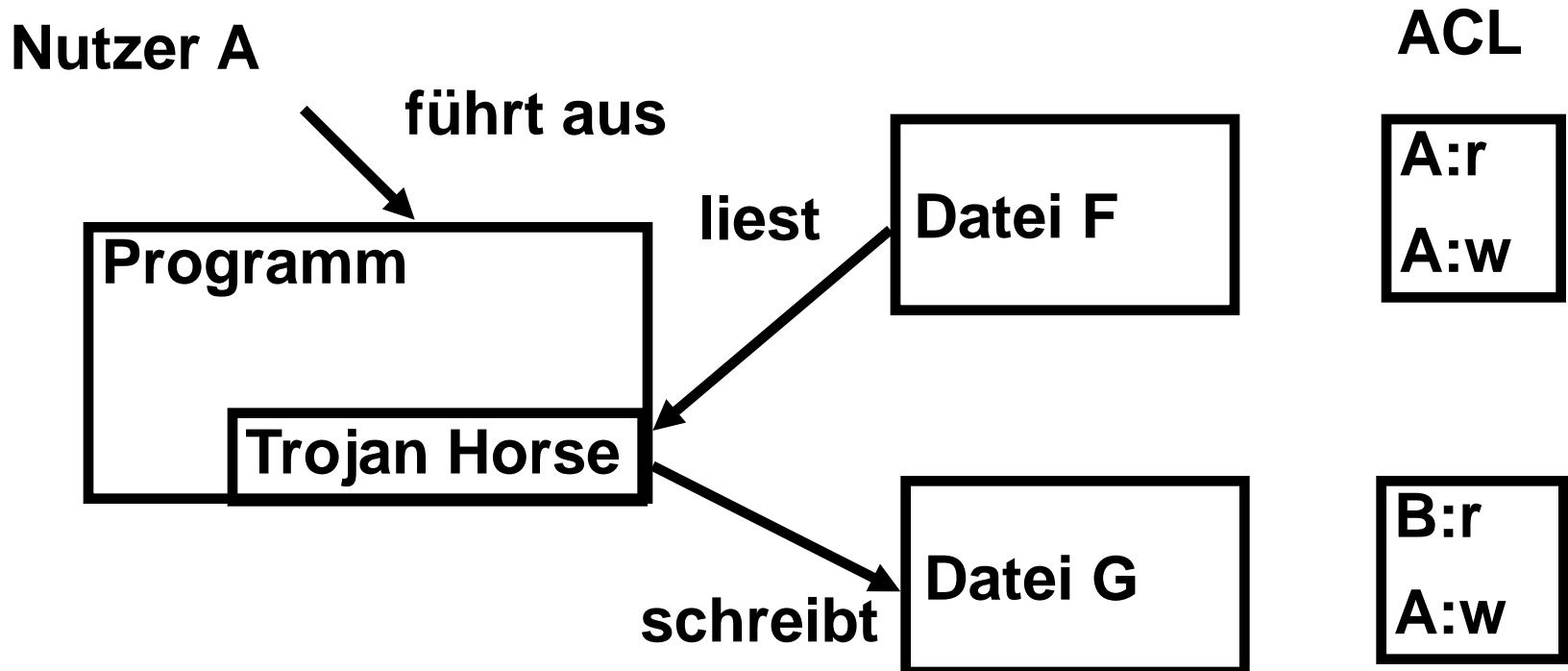
Datei G

B:r  
A:w

Nutzer B kann Datei F nicht lesen

# TROJANISCHE PFERD: BEISPIEL

Nutzer B erstellt/modifiziert Programm mit einem trojanischen Pferd, das von Nutzer A ausgeführt werden kann



Nutzer B kann den von F nach G kopierten Inhalt lesen

# SYSTEMBESTIMMTE ZUGRIFFSKONTROLLE

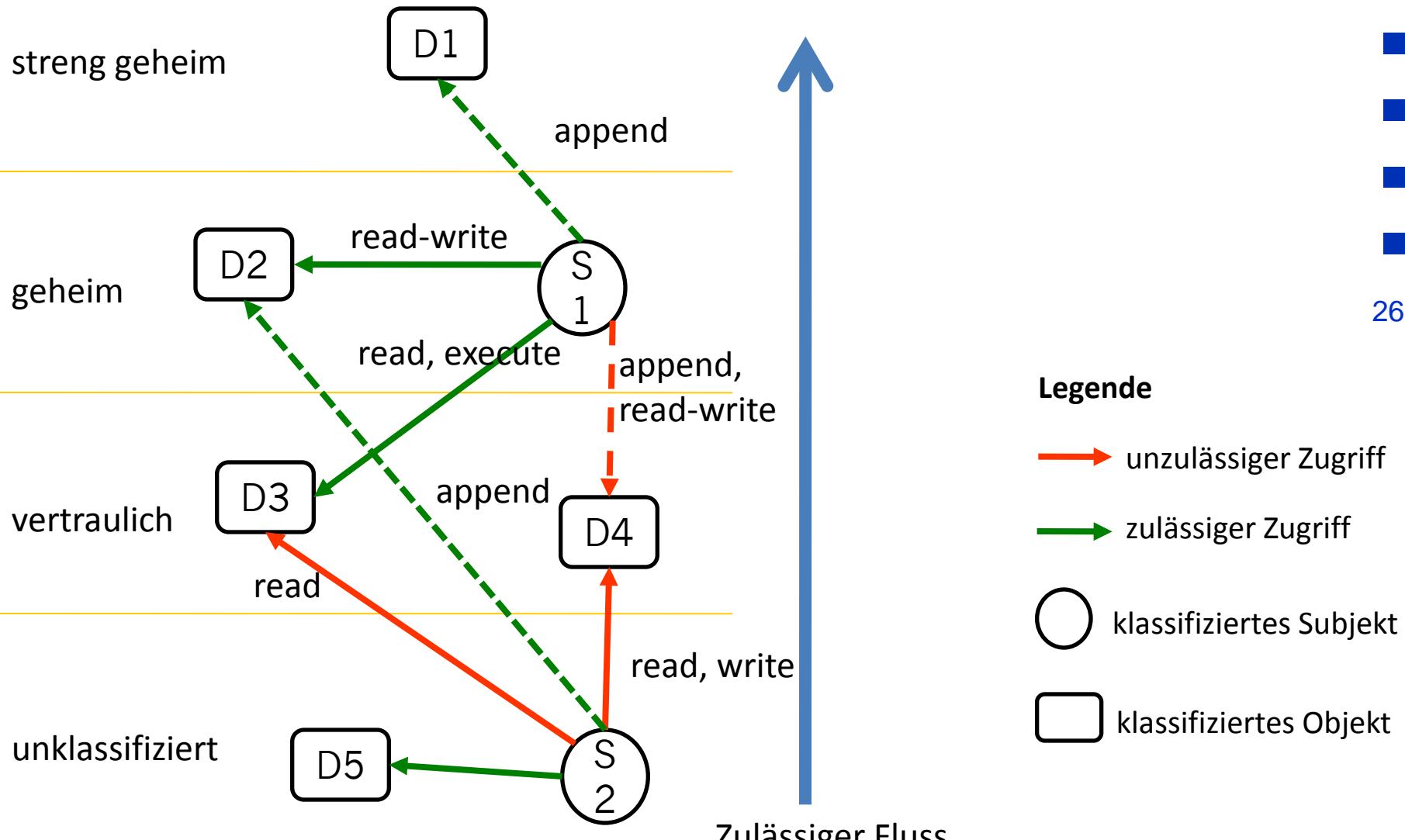
- (Mandatory Access Control – MAC)
- System gewährt Zugriffsrechte gemäß systemweiter Richtlinie
- Nutzern wird die volle Kontrolle über die Zugriffsrecht der von ihnen angelegten Ressourcen verwehrt
- Beispiele:
  - Bell/LaPadula
  - Biba
  - Chinese-Wall

24

# BELL-LA PADULA-MODELL

- David Bell und Len La Padula, 1973 auf Initiative der US Air Force entwickelt
- Zweck:
  - Fokussiert Vertraulichkeit von Information
  - Einfacher Sicherheitsmechanismus, der Verifikation erlaubt
  - Ermöglicht formalen Nachweis, dass Computer System klassifizierte Information sicher verarbeiten kann
- Multi-Level Security System (MLS)
  - Klassifikation von Objekten hinsichtlich Grad der Vertraulichkeit, z.B.: streng geheim, geheim, vertraulich, öffentlich/unklassifiziert
  - Klassifikationen von Subjekten hinsichtlich Freigabe für Vertraulichkeitsklassen
  - Information kann nicht nach unten fließen, z.B. von streng geheim nach öffentlich
- Statisches Modell: keine Änderung der Vertraulichkeitsstufen /Security Level

# BELL-LA PADULA-MODELL



## Systembestimmte Zugriffsbeschränkungen

### ■ No-read-up

- Eine Lese- oder Executezugriff auf ein Objekt o ist nur zulässig, wenn s das entsprechende Zugriffsrecht r besitzt und die Objektklassifikation kleiner oder gleich der Subjekt-Freigabe.

### ■ No-write-down

- Ein Append-Zugriff auf ein Objekt o durch eine Subjekt s ist nur zulässig, wenn die Objektklassifikation mindestens so hoch ist, wie die Freigabe des Subjekts und ein Lese-Schreib-Zugriff ist nur zulässig, wenn die Objektklassifikation gleich der Freigabe des Subjekts ist.

# BELL-LA PADULA-MODELL - GRENZEN

- Problem: Informationen werden sukzessive immer höher eingestuft.  
Warum tritt das Problem auf?
  - Sachbearbeiter erstellt eine Dokument; Chef korrigiert das Dokument; Chef-Chef korrigiert das Dokument. Damit Chef Dokument schreiben kann, muss es entsprechend seiner Freigabe in der Klassifikation angehoben werden.
- Problem des Blinden Schreibens: darf o modifizieren aber anschließend (wegen no-read-up) nicht lesen  
Problem?
  - Versehentliche oder mutwillige Integritätsverletzungen.
- Betrachtet nur Vertraulichkeit

# BIBA MODELL

- Betrachtet Integrität
- Klassifikation gemäß Integritätsstufen
  - Höhere Stufen: vertrauenswürdiger
- Entsprechung des Bell-La Padula-Modells für Integrität
- Subjekt und Objekte besitzen Integritäts-Level

29

## Systembestimmte Zugriffsbeschränkungen

- $i(s)$  = Integritätslevel von Subjekt s
- $i(o)$  = Integritätslevel von Objekt o
- No-write-up
  - Subjekt s kann Objekt o schreiben, wenn  $i(o) \leq i(s)$
  - Subjekt kann nicht in höhere Integritätslevel als sein eigener schreiben.
- No-read-down
  - Subjekt s kann Objekt o lesen, wenn  $i(o) \geq i(s)$
  - Subjekt kann nicht durch weniger vertrauenswürdige Daten beeinflusst werden

# CHINESE WALL MODELL

- Brewer-Nash, 1989
- Ziel: unzulässige Ausnutzung von Insiderwissen bei
  - Bank- und Börsentransaktionen oder
  - Beratung von Unternehmen verhindern.
- Idee: zukünftige Zugriffsmöglichkeiten eines Subjekts werden durch die Zugriffe, die Subjekt in der Vergangenheit getätigt hat, beschränkt.
  - Zugriffe sind abhängig von der Zugriffshistorie



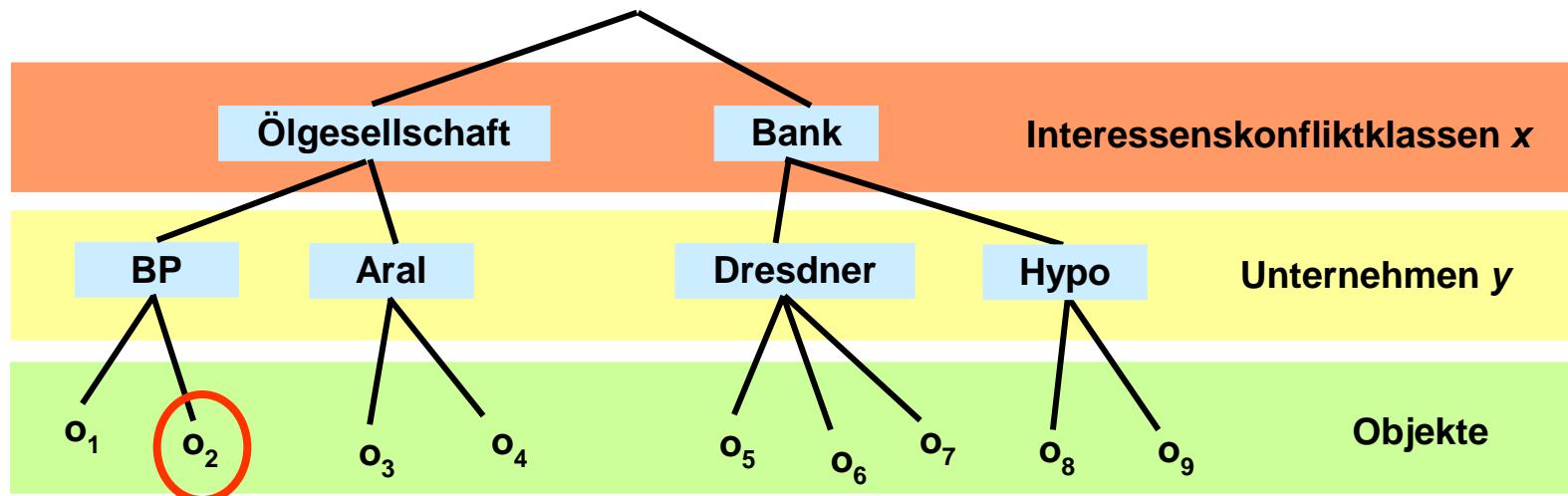
31

# CHINESE WALL MODELL

- Interessenkonflikt-Klassen
- Zugriffe sind abhängig von der Zugriffshistorie

COI (Conflict of Interest)-Regel:

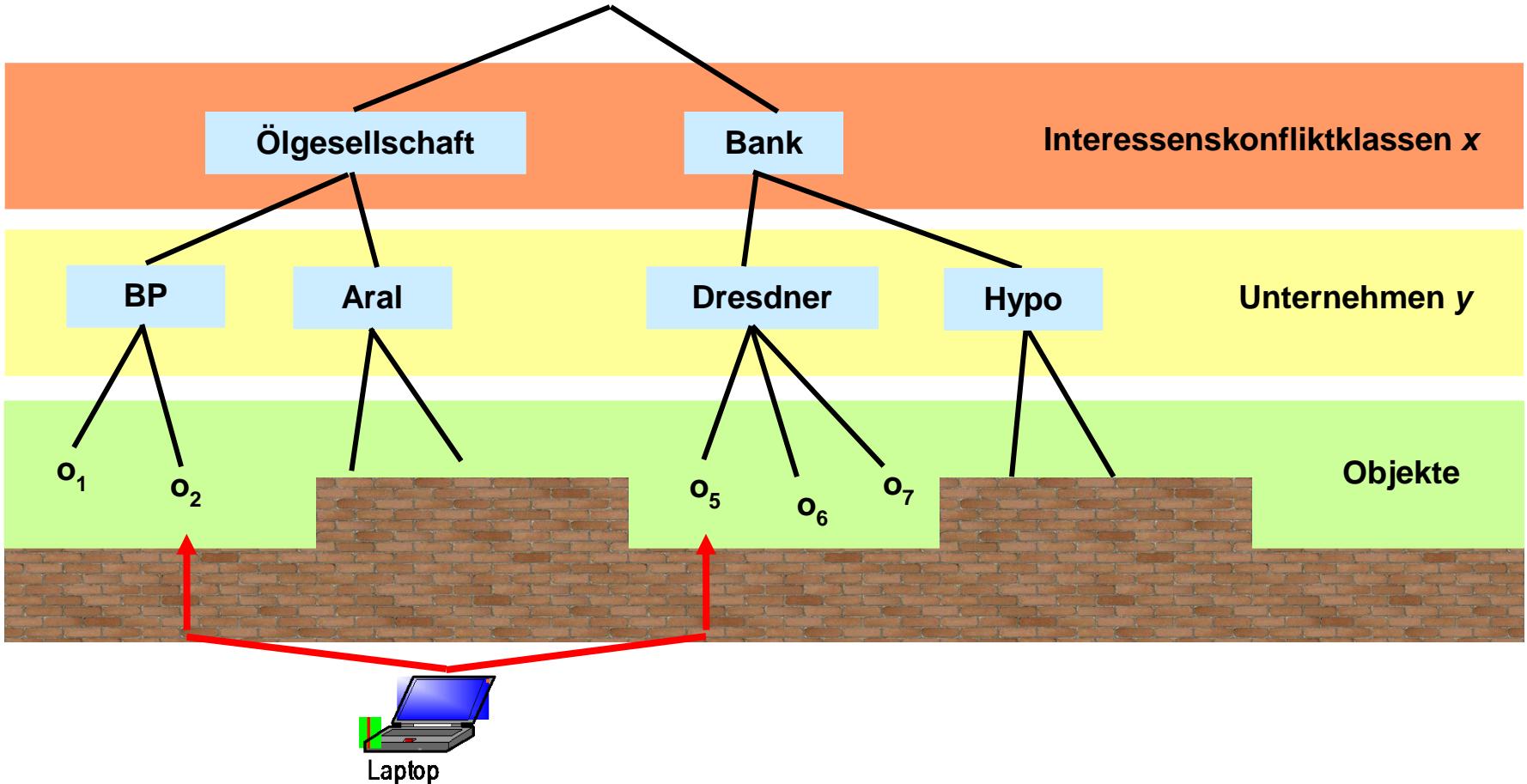
- Nach einem Zugriff auf Objekte eines Unternehmens U darf nicht mehr auf Objekte eines Unternehmens V ( $\neq U$ ) zugegriffen werden, wenn die Unternehmen im Konflikt zueinander stehen.



- Zugriff: Subjekt  $s_1$  auf das Objekt  $o_2$  (zum Zeitpunkt  $t'$ )

# CHINESE WALL MODELL

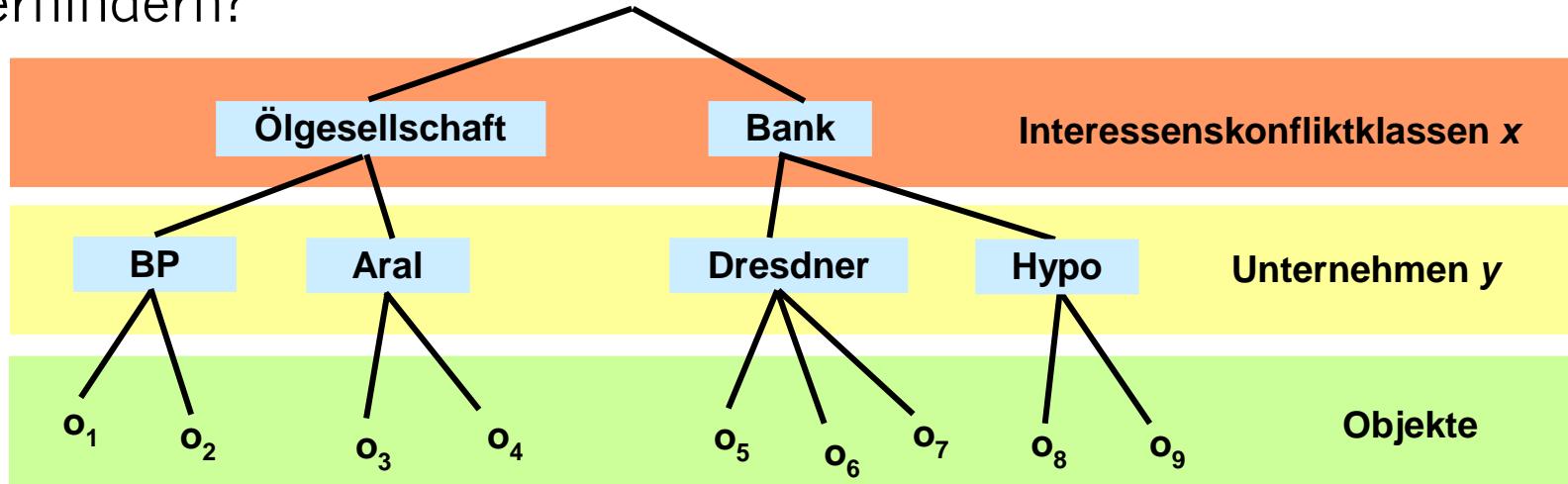
- Nach Zugriff (kontext-abhängig) wird eine Zugriffsmauer gebaut
- Beispiel: Zugriffsmauer nach dem Zugriff auf  $o_2$  und  $o_5$



# CHINESE WALL MODELL

## Regeln

- Lesezugriffregel:
  - nur wenn **keine** früheren Zugriffe auf andere Objekte aus der selben Konfliktklasse aber aus anderen Unternehmen
- Genügt diese Regel um unerwünschten Informationsfluss zu verhindern?



- Zwei Subjekte  $s_1$  und  $s_2$ 
  - $s_1$  liest  $o_1$  und schreibt  $o_5$ ;  $s_2$  liest  $o_5$  und schreibt  $o_3$
  - Informationsfluss von  $o_1$  nach  $o_3$

## Regeln

- Schreibzugriffsregel:
  - Nur wenn sich alle früheren Lesezugriffe auf Objekte desselben Unternehmens oder öffentliche Objekte bezogen

# CHINESE WALL MODELL

## Probleme

- Aufhebung von Interessenkonflikten nicht vorgesehen
- Zugriffshistorie wächst im Prinzip unbegrenzt

37

# FRAGEN?

38

- Subjekte
  - Benutzer
  - Benutzergruppen
  - Prozesse
- Objekte
  - Dateien
    - (externe) Geräte wie Monitor, Drucker, Modem sowie Massenspeicher wie Festplatten, CD-Rom-Laufwerke oder auch Arbeitsspeicher des Betriebssystemkerns (`/dev/kmem`) werden als Dateien modelliert
  - Verzeichnisse

# IDENTIFIKATION

- von Dateien
  - Baumartig strukturiertes Dateisystem
  - Benennung einer Datei oder eines Verzeichnisses durch Pfadnamen
- von Prozessen
  - pid - vom Betriebssystemkern vergebene eindeutige Prozessidentifikatoren
- von Benutzern und Gruppen
  - uid und guid – eindeutige User/Group Identifikatoren
  - uid = 0 vorgesehen für **Superuser**
  - unterschieden werden **reale** und **effektive** uid und guid
  - Zugriffsberechtigungen hängen von der effektiven uid und guid ab
  - reale uid und guid fest
  - effektive uid und guid kann sich dynamisch im Verlauf einer Sitzung ändern (vgl. setuid-Konzept)

41

# IDENTIFIKATION

- „Namen sind irrelevant“
- uids werden in /etc/passwd Benutzernamen zugeordnet
- guids werden in /etc/group Gruppennamen zugeordnet
- Für Rechteüberprüfung ist **nicht** der Benutzername und **nicht** der Gruppename relevant sondern nur die uid und die guid
- Benutzer mit uid = 123 und Benutzername root ist keineswegs Superuser
- Superuser: uid = 0

42

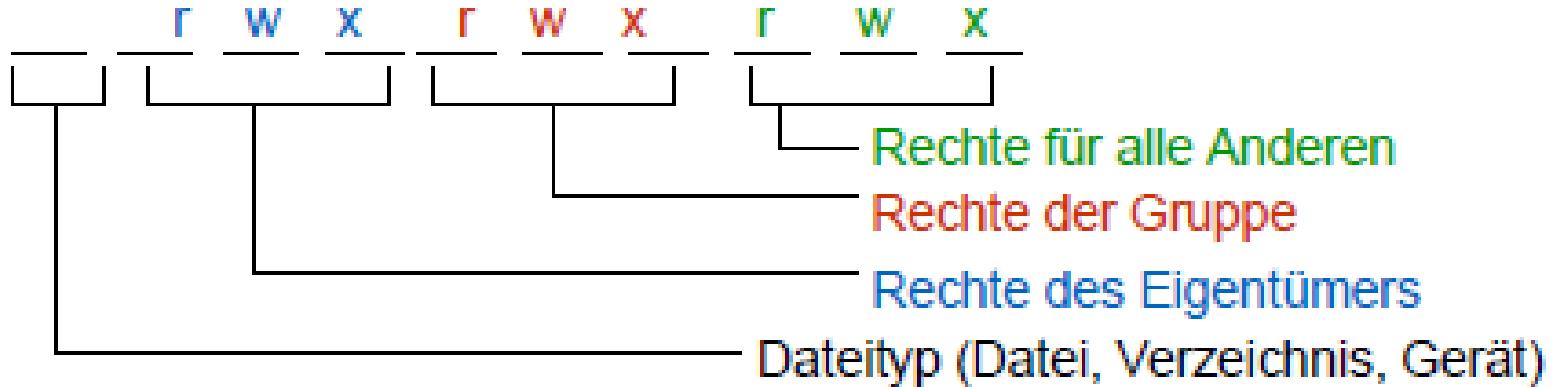
# RECHTEVERGABE

- (im Wesentlichen) verfügbare Zugriffsrechte
  - r - lesen
  - w - schreiben
  - x - ausführen
- nur der Eigentümer des Objekts o und Superuser können die Rechte an Objekt o ändern (also hinzufügen oder entfernen)
- Jedes Objekt besitzt eine Zugriffskontrollliste mit Rechten für drei Klassen von autorisierten Benutzern:
  - Objekteigentümer (Owner-Rechte)
  - Benutzer, die der Gruppe angehören, der das Objekt gehört (Gruppen-Rechte)
  - alle anderen Benutzer (World-Rechte)

43

# UNIX-ZUGRIFFSKONTROLLLISTE

## ■ Allgemeine Struktur



- An Dateitypen werden unterschieden
  - einfache Datei (regular file)
  - d Verzeichnis (directory)
  - l Verweis (link)
  - c zeichenorientiertes Gerät (z.B. Terminal, Drucker)
  - b blockorientiertes Gerät (z.B. Band)

```
-rw----- 1 meier issi 81920 Dec 21 2007 testfile.tar
```

45

# ZUGRIFFSRECHTE VERÄNDERN MIT CHMOD

- Rechte für Dateieigentümer (User) ändern  
`chmod u+r datei`  
`chmod u-r datei`
- Rechte für die Gruppe ändern, zu der Datei/Verzeichnis gehört  
`chmod g+w datei`
- Rechte für alle Anderen (Other) ändern  
`chmod o+x datei`

46

# RECHTE-SEMANTIK FÜR VERZEICHNISSE

- r – Leserecht: erlaubt den Inhalt des Verzeichnisses aufzulisten (z.B. mit ls)
- w – Schreib-Recht: erlaubt das Hinzufügen oder Entfernen von Elementen zu bzw. aus dem Verzeichnis
- x – Suchrecht: erlaubt das Verzeichnis als Teil eines Pfadnamen zu durchlaufen bzw. das Verzeichnis mit dem Kommando cd „zu betreten“ und darin befindliche Dateien zu öffnen
- Beispiel: drwx r-- ---
  - Objekt ist Verzeichnis
  - Eigentümer besitzt Lese-, Schreib- und Ausführungs-/Such-Recht
  - Mitglieder der Gruppe, zu der das Objekt gehört, haben das Recht das Verzeichnis zu lesen, also den Inhalt aufzulisten
  - Allen anderen Benutzern sind alle Zugriffe verwehrt.

47

# SUCH-RECHT FÜR VERZEICHNISSE

- Um auf eine Datei zugreifen zu können muss der zugreifende Nutzer für alle Verzeichnisse des Pfadnamens der Datei das Such-Recht besitzen
  - /home/meier/Windows/LehreDo/bilder.doc
  - Um auf bilder.doc zugreifen zu können, müssen für Verzeichnisse home, meier, Windows und LehreDo x-Rechte vorliegen
- Suchrecht ist **nicht** gleichzeitig mit Leserecht verknüpft, sodass allein mit x-Recht ein Auflisten des Verzeichnisinhalts **nicht** zulässig ist.
- Idee: Nur wer Dateinamen im Verzeichnis kennt, kann auf Datei zugreifen.  
⇒ Verstecken von Dateien, allerdings nur schwacher Schutz

# SCHREIB-RECHT FÜR VERZEICHNIS

- Mit w- und x-Recht für ein Verzeichnis ist das Recht zum Entfernen von Dateien aus dem Verzeichnis erteilt, unabhängig davon, wie die Rechte der Dateien gesetzt sind und unabhängig davon welche Benutzer Eigentümer der Dateien sind.

49

# BEISPIEL

- Datei f mit

`rw- r-- ---`

Kann ein Mitglied der Gruppe, zu der die Datei gehört, die Datei verändern?

- Datei f befindet sich in **Verzeichnis d**, das zur gleichen Gruppe wie Datei f gehört, mit

`rwx -wx ---`

- Alle Benutzer der Gruppe, zu der das Verzeichnis gehört, besitzen Schreibrecht für Verzeichnis d, jedoch nicht das Recht auf die Datei f zu schreiben.
- Schreibrecht für Verzeichnis d impliziert Löschrecht für Datei f
- Benutzer der Gruppe können Datei f entfernen und durch neue Datei f' ersetzen!
  - Diese Zugriffe sind durch Rechte für Verzeichnis d erlaubt widersprechen jedoch der intendierten Rechtevergabe für Datei f

- Detailkenntnisse erforderlich um Inkonsistenz zu vermeiden.

# SONDERRECHTE – STICKY-BIT

- Ursprünglich für Dateien vorgesehen, um zu signalisieren, dass Datei im Speicher zu halten ist, also von der Speicherverwaltung nicht auf Festplatte ausgelagert werden darf.  
Sticky-Bit heute nur noch für Verzeichnisse relevant.
- Jeder Benutzer mit Schreibberechtigung für ein Verzeichnis hat die Berechtigung jede Datei in dem Verzeichnis zu löschen, auch Dateien, deren Eigentümer er **nicht** ist.
- Gesetztes Sticky-Bit schränkt Löschrecht auf den Eigentümer der zu löschen Datei und Eigentümer des Verzeichnisses (und Superuser) ein.
- Insbesondere sinnvoll für Verzeichnis /tmp für das alle Benutzer Schreib- und Ausführungsrechte besitzen.



```
drwxrwxrwt 6 sys sys 577 Dec 2 11:15 tmp
```

# SONDERRECHTE – SETUID/SUID

- uid-Konzept ermöglicht die temporäre Weitergabe von Rechten eines Benutzer
- Führt ein Benutzer eine Datei mit gesetztem uid-Bit aus, so wird die effektive User-ID des ausführenden Prozesses, auf die User-ID des Eigentümers der Datei gesetzt.
  - Der Prozess führt das Programm mit den Rechten des Eigentümers aus.
- Suid-Bit kann vom Eigentümer oder Superuser gesetzt werden  
`chmod u+s datei`  
Voraussetzung: Ausführungsrechte für Gruppe oder Other
  - Nur für Dateien, nicht für Verzeichnisse.
- Wo benötigt man das?



```
-r-sr-sr-x    1  root      sys     ...   /bin/passwd
```

# SONDERRECHT – SETGID/SGID

- Analog SUID
- sgid-Konzept ermöglicht die temporäre Weitergabe von Rechten einer Gruppe
- Führt ein Benutzer eine Datei mit gesetztem sgid-Bit aus, so wird die effektive Group-ID des ausführenden Prozesses, auf die Group-ID der Eigentümergruppe der Datei gesetzt.
  - Der Prozess führt das Programm mit den Rechten der Eigentümergruppe aus.

53

# RISIKEN VON SUID

## ■ Beispiel 1:

-rwsr-rwx klaus stud ... beispielprogramm

- Schreibbar und ausführbar für Alle
  - ⇒ Jeder kann Programm ändern!
- Läuft aufgrund SUID mit den Rechten von klaus
  - ⇒ Jeder kann beliebige Aktionen mit den Rechten von klaus ausführen!

54

## ■ Beispiel 2:

Der Superuser lässt seinen Rechner einen Moment unbeaufsichtigt:

- Angreifer tut folgendes:

```
cp /bin/sh /tmp/endlich_root  
chmod o+x /tmp/endlich_root  
chmod +s /tmp/endlich_root
```

- ⇒ Kommandozeilen-Interpreter/Shell die mit Root-Rechten ausgeführt wird!

# SONDERRECHT – SETGID/SGID FÜR VERZEICHNISSE

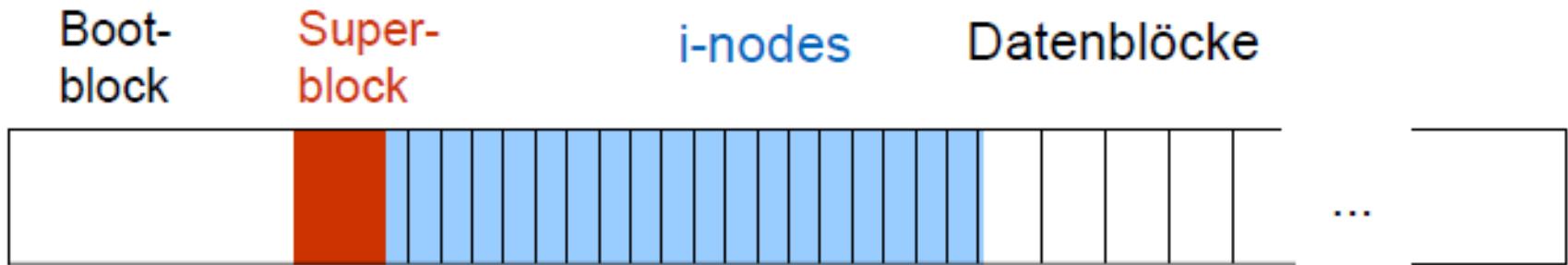
- Erstellt ein Benutzer eine neue Datei in einem Verzeichnis dann ist dieser Benutzer Eigentümer der Datei und die Datei gehört zur (primären) Gruppe des Benutzers
- Wird eine neue Datei in einem Verzeichnis erstellt für das SGID gesetzt ist, dann gehört die neue Datei zu der Gruppe, zu der das Verzeichnis gehört.
- Wo benötigt man das?
  - Wenn Nutzer mehreren Gruppen zugehören und Dateien untereinander teilen

55

# UMSETZUNG DER ZUGRIFFSKONTROLLE

- Dateien und Verzeichnisse werden Betriebssystem-intern über einen Datei-Deskriptor, den **i-node** (index-node), beschrieben
- **i-node** (64 Byte) enthält u.a. Namen des Datei-Eigentümers und die ACL
- **i-nodes** werden auf der Festplatte verwaltet
- Beim Öffnen einer Datei wird **i-node** geladen

56

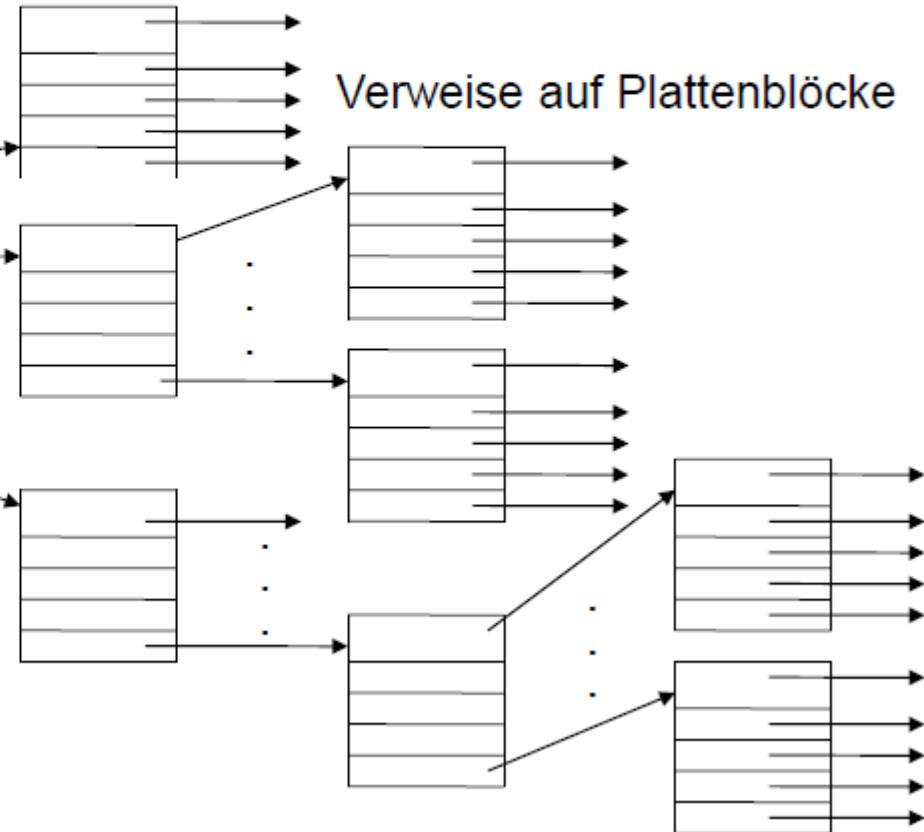


# I-NODE

owner joe, uid  
group student, guid  
type regular file  
**perms rwxr-xr-x**  
accessed Feb 12 1999 3:00 P.M.  
modified Feb 11 1999 10:16 A.M.  
Adressen der 10 ersten Plattenblöcke  
einfach indirekt  
zweifach indirekt  
dreifach indirekt

ACL als Bestandteil der i-node unter Unix

Unix-i-node



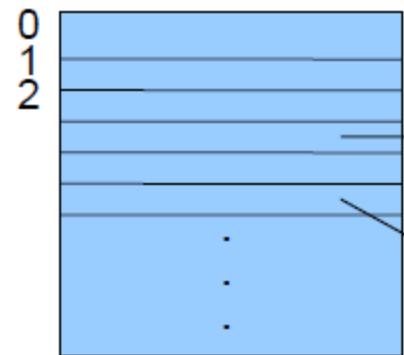
# ABLAUF DER ZUGRIFFSKONTROLLE

- Open-System-Call: Angabe des Zwecks r, w, x
- Aktionen im Betriebssystem-Kern (siehe nächste Folie)
  - (1) Laden des i-node der zu öffnenden Datei in i-node Tabelle der Kernels
  - (2) Prüfen, ob zugreifender Prozess gemäß ACL der Datei zum gewünschten Zugriff r, w, x berechtigt ist
  - (3) Falls ok, return File-Handle: enthält Information über zulässige Zugriffsrechte r, w, x
    - Eintrag mit Rechten in Open-File-Tabelle des Kernels
    - Verweis in File-Descriptor-Tabelle des Prozesses auf Recht
  - (4) Zugriffe auf geöffnete Datei mit File-Handle  
Dateisystem führt Zulässigkeitskontrolle durch.

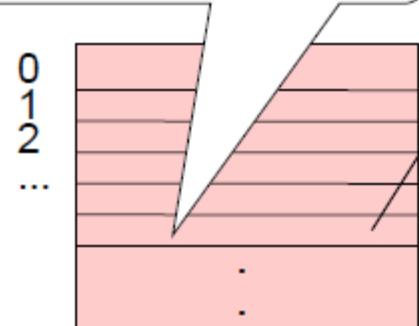
58

# DATENSTRUKTUREN IM KERN

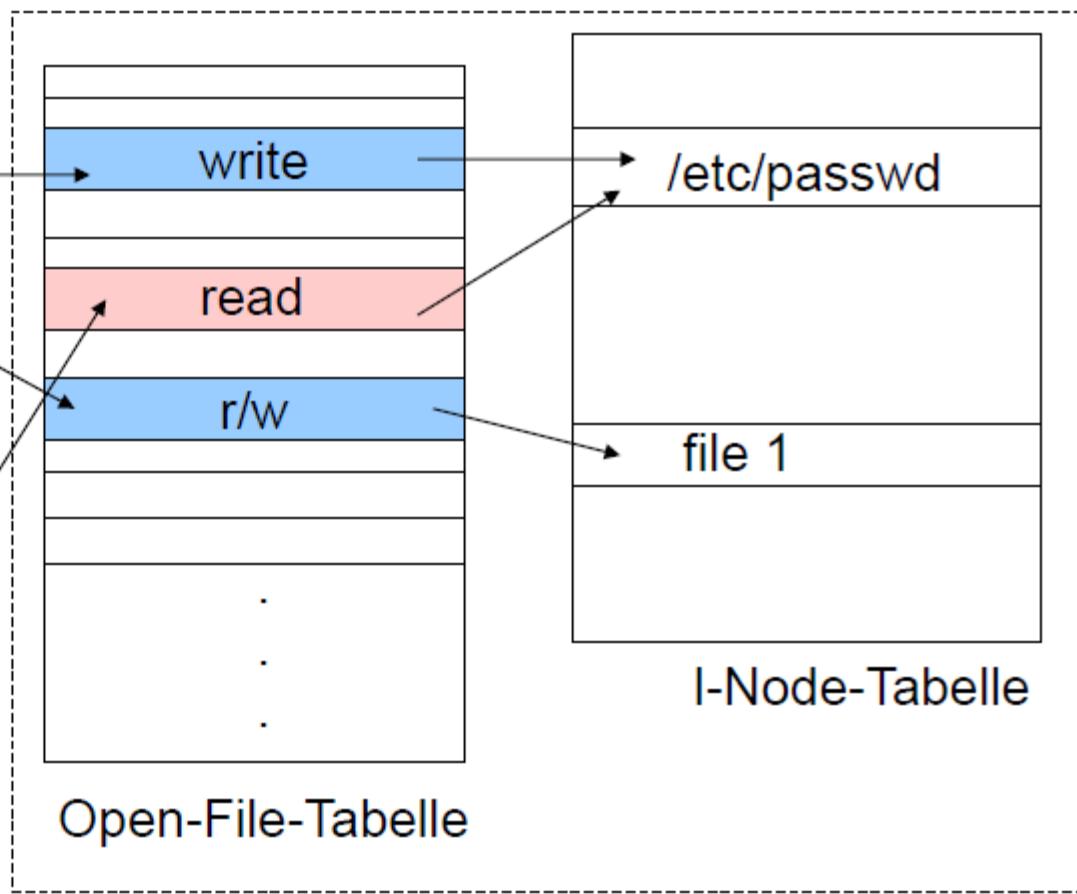
Prozess A  
File-Descriptor-Tabelle für A



File-Handle zur lesenden Nutzung  
von /etc/passwd durch Prozess B



BS-Kerndatenstrukturen für Unix-Zugriffskontrolle:



(BS)-systemglobale Tabellen

# FAZIT

- Einfaches Modell für benutzerbestimmbare Zugriffskontrolle
  - (Hat dennoch bereit seine Tücken)
- Kombination aus
  - objektspezifischen Zugriffskontrolllisten und
  - subjektspezifischen Zugriffsausweisen
- Rechterücknahme hat keine unmittelbare Auswirkung auf die Prozesse, die Datei in geöffnetem Zustand halten
- Verwendete ACLs erlauben nur geringe Differenzierung bei Rechtevergabe
  - i.d.R. werden zu viele Rechte vergeben (anstatt least privilege)
- Superuser-Konzept – vollständige Rechtekonzentration (anstatt separation of duty)
- Risiken bei der Rechtedelegation mit dem SetUID-Konzept

60

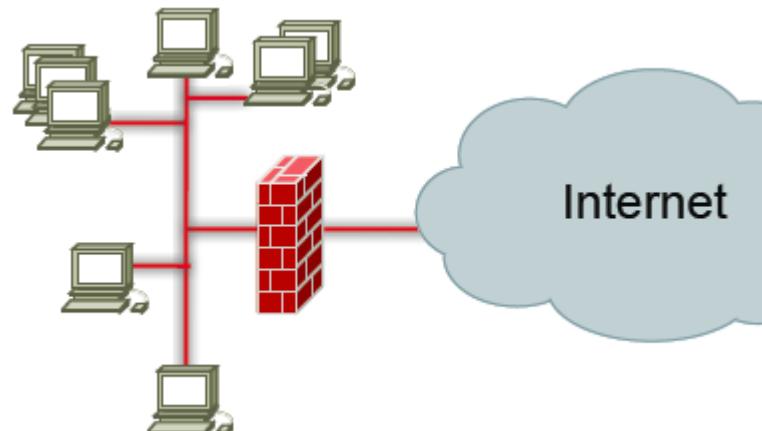
# FRAGEN?

61

# ZUGRIFFSKONTROLLE IM NETZ MIT FIREWALLS

# FIREWALLS

- Notwendigkeit der Abgrenzung zwischen Netzen verschiedener Vertrauenswürdigkeit und Sicherheitsanforderungen
- kontrollierte Durchlässigkeit an Netzgrenzen durch Zugriffskontrolle (Firewall)
- Firewall
  - Zweck
    - Abschottung eines Intranets bzw. eines sicherheitsempfindlichen Teilnetzes vor unberechtigten Zugriffen von außen
    - Alle Datenpakete zum und vom Intranet müssen Firewall passieren
    - Regelbasierte Filtersysteme
  - besteht aus Hard- und Software, die zwei oder mehr Netze koppelt und durchsetzt, dass jeglicher Verkehr zwischen den Netzen durch die Firewall geleitet wird.
  - realisiert eine Sicherheitspolitik:
    - Zugriffsrestriktionen für Paketweiterleitung
    - Protokollierungsanforderungen
    - Authentifizierungsanforderungen



# TYPEN VON FIREWALLS

- Paketfilter
  - überprüfen anhand der IP-Adresse des Absenders und Empfängers sowie Portnummer (TCP oder UDP), ob Datenpakete die Firewall passieren dürfen
  - teilweise auch Analyse des Inhalts der Datenpakete
- Circuit Level Gateway
  - Überprüfen TCP- und UDP-Verbindungen
  - Verbindung: viele Datenpakete
  - Firewall ersetzt bei gehenden Verbindungen die ursprüngliche Absenderadresse durch die eigene IP-Adresse und verbirgt somit die interne Netzstruktur
- Application Level Gateways (auch: Proxies)
  - Implementieren die Schnittstelle sowohl des Clients als auch des Servers eines Dienstes

64

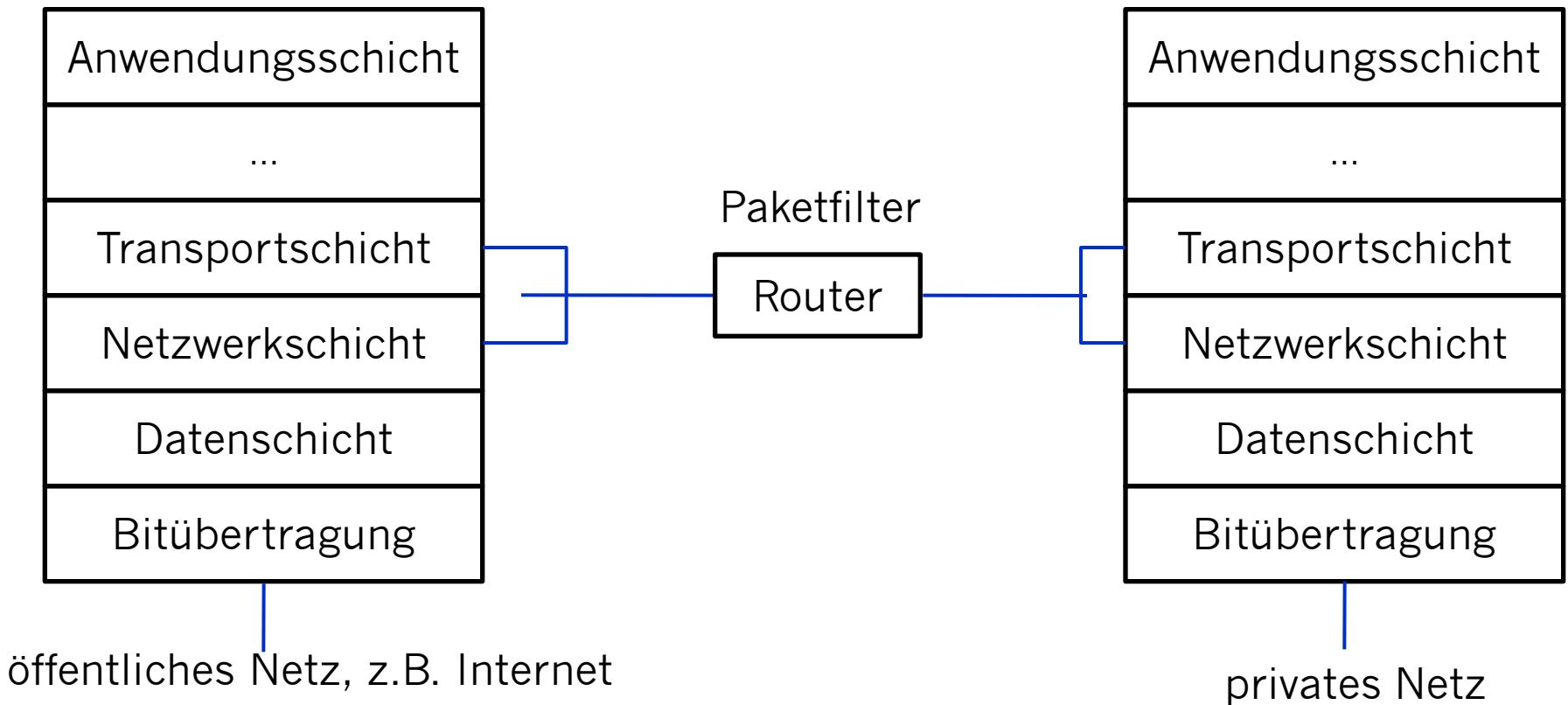
# MERKMALE VON FIREWALLS

- Paketfilter und Circuit Level Gateways
  - können transparent für Endbenutzer eingesetzt werden
  - Anwender im Intranet muss von Existenz nichts mitbekommen
- Proxies
  - Anwendungsunterstützung notwendig: Anwendungen müssen proxytauglich sein
  - existieren für fast alle relevanten Anwendungen (Web, Filetransfer, News)
  - Problem:
    - für jede Anwendung (bzw. deren Protokoll) muss eigener Proxy vorhanden sein
  - Lösung:
    - SOCKS-Proxy
      - universelle Proxy-Schnittstelle (anwendungsunabhängig)

65

# PAKETFILTER

- Einordnung in die ISO/OSI-Schichten



# BEWERTUNG PAKETFILTER

## Positiv

- Einfach und preiswert
- Effizient
- Gut mit Router-Funktionalität kombinierbar (Screening Router)

## Negativ

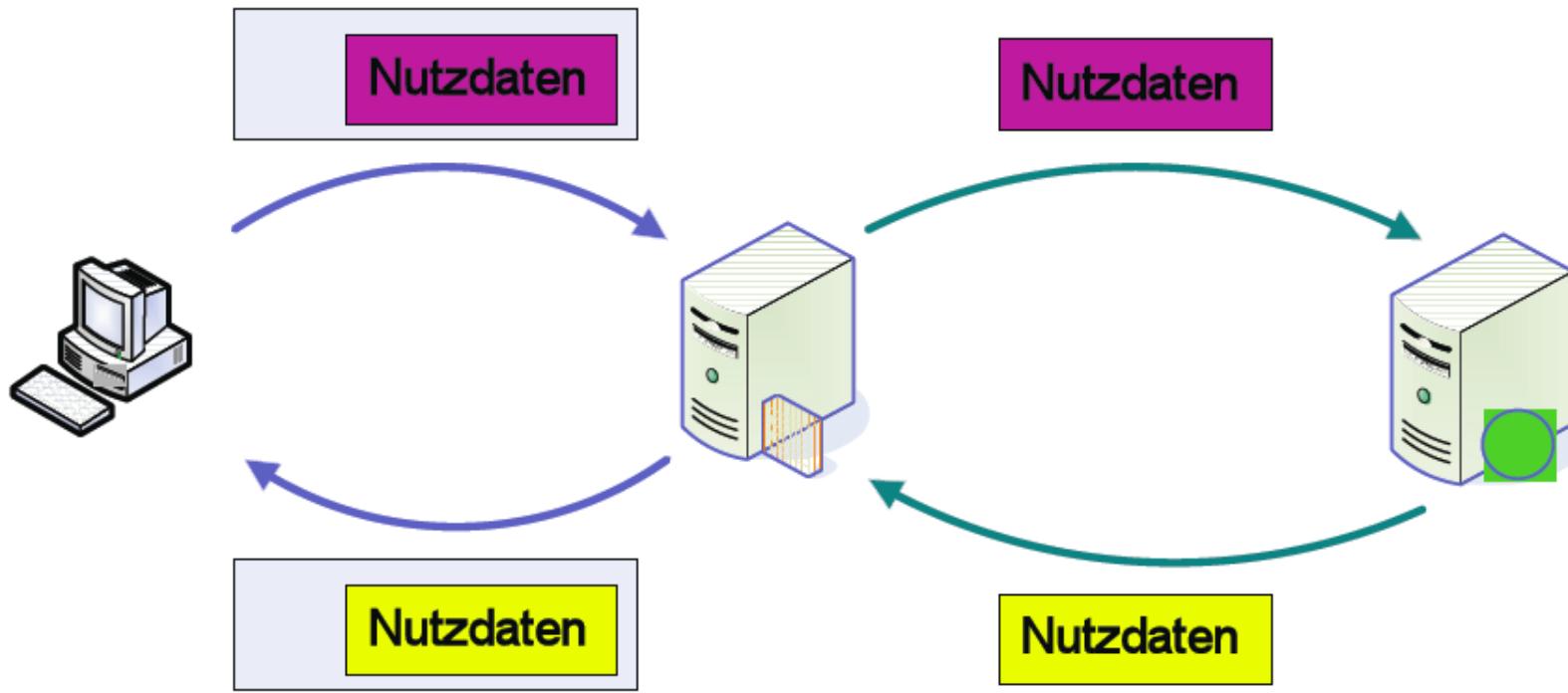
- Integrität der Header-Informationen nicht gesichert; alle Felder können relativ einfach gefälscht werden
- Grobgranulare Filterung
- Keine inhaltliche Analyse bei freigegebenen Diensten
- Abbildungsproblematik: Policy (Prosa) auf konkrete FW-Regeln
- Erstellung einer Filtertabelle nicht triviale Aufgabe
  - Korrektheit ?
  - Vollständigkeit ?
  - Konsistenz ?

67

# CIRCUIT LEVEL GATEWAY (VERBINDUNGS-GATEWAY)

- Circuit Level Gateway (CLG) stellt generischen Proxy dar
  - CLG auch als „Multiprotokollproxy“ bezeichnet
- Nicht auf einzelne Dienste zugeschnitten, allgemeiner „Vermittler“ von Verbindungen
- Trennt Verbindung zwischen Client und Server
- Benutzeroauthentifikation am Gateway möglich
- Bsp. SOCKS:
  - SOCKS-Server filtert den TCP/IP Verkehr
  - Alle Verbindungen der Clients müssen über SOCKS-Server laufen
  - Daher Anpassung der Clients erforderlich („socksifying“)
  - Filtert nach: Quelle, Ziel, Art des Verbindungsaufbaus (z.B. Initiierung oder Antwort), Protokoll, Benutzer

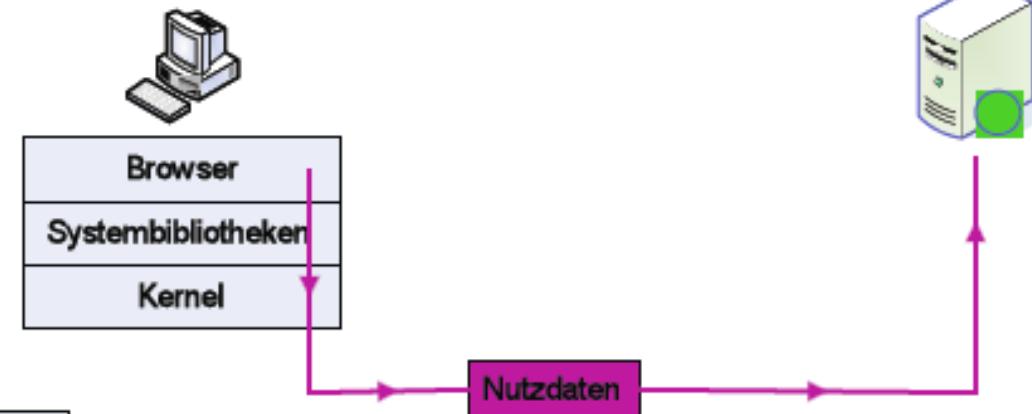
# PRINZIPIELLER ABLAUF BEIM EINSATZ VON SOCKS



- Systemfunktionen wie `connect()` und `bind()` laufen über den SOCKS-Server
- Damit auch für Server, nicht nur für Clients geeignet

# „SOCKSIFY“ MIT LD\_PRELOAD UNTER LINUX

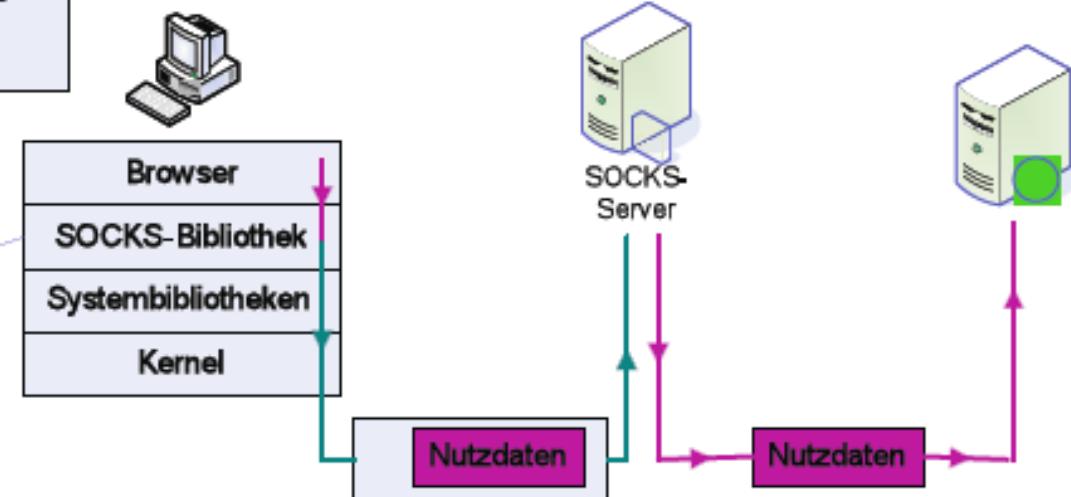
Ohne „Socksifizierung“:



```
export SOCKS_USERNAME="secp"  
export SOCKS_PASSWORD="geheim"  
export LD_PRELOAD="/usr/lib/libdsocks.so"  
firefox
```

Mit „Socksifizierung“:

*Funktioniert nicht bei  
„statisch“ gelinkten Binaries*



# BEWERTUNG CIRCUIT LEVEL GATEWAY

## Positiv

- Anwendungsunabhängige Filterung
- Ein Proxy für alle Dienste
- Umfangreiche Logging Möglichkeit und damit Accounting
- Zustandsbehaftet
- Benutzerauthentifikation und benutzerabhängige Filterung
- Entkopplung von internem und externem Netz
- Möglichkeit der Erstellung von Nutzungsprofilen

71

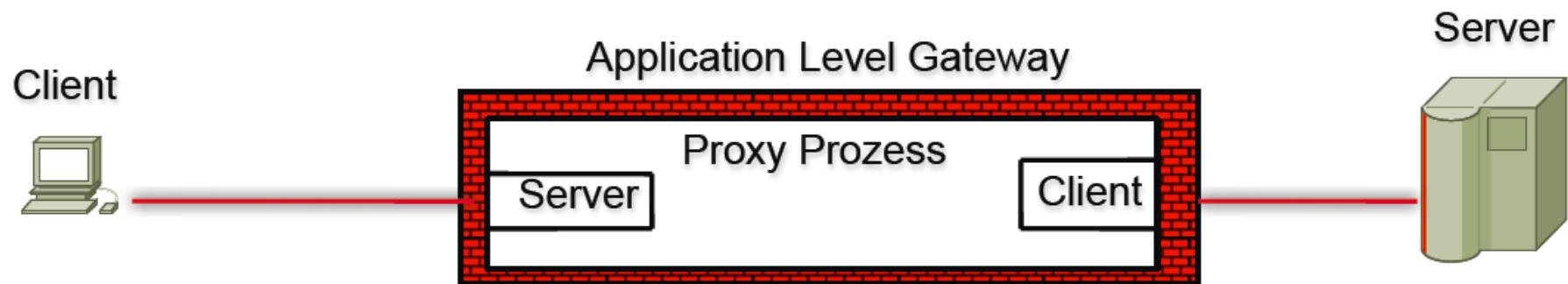
## Negativ

- Möglichkeit der Erstellung von Nutzungsprofilen
- I.d.R. keine Filterung nach Dienstprimitiven möglich
- Sicherheit der Proxy-Implementierung und -Konfiguration?
- Support durch oder Modifikation der Clients erforderlich

# APPLICATION LEVEL GATEWAY (ALG, APPLIKATIONSFILTER)

- Filtern auf Schicht 7 (Anwendungsschicht)
- Analyse des Anwendungsschichtprotokolls u. d. Nutzdaten
- Für **jeden** Dienst, jedes Protokoll ist ein eigener Filterprozess (auch als **Proxy** bezeichnet) erforderlich
- Interner Client muss sich i.d.R. am Proxy authentifizieren
- Proxy trennt Verbindung zwischen Client und Server
- Nach außen erscheint immer nur die Adresse des Application Level Gateways; völlige Entkoppelung von internem und externem Netz
- ALG kann Zustandsinformationen halten und nutzen

72



# ABLAUF EINER VERBINDUNG ÜBER ALG



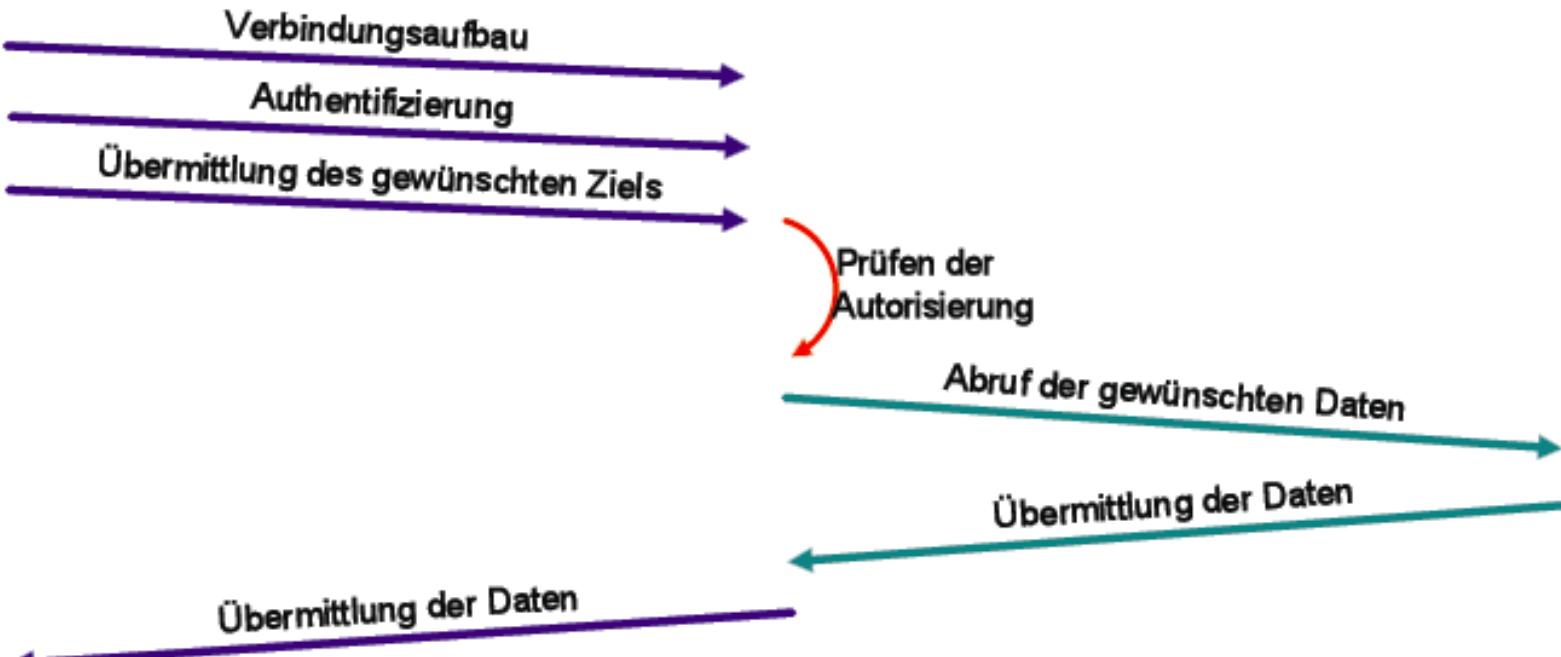
Client:  
Browser



Application  
Level Gateway  
für HTTP



Webserver



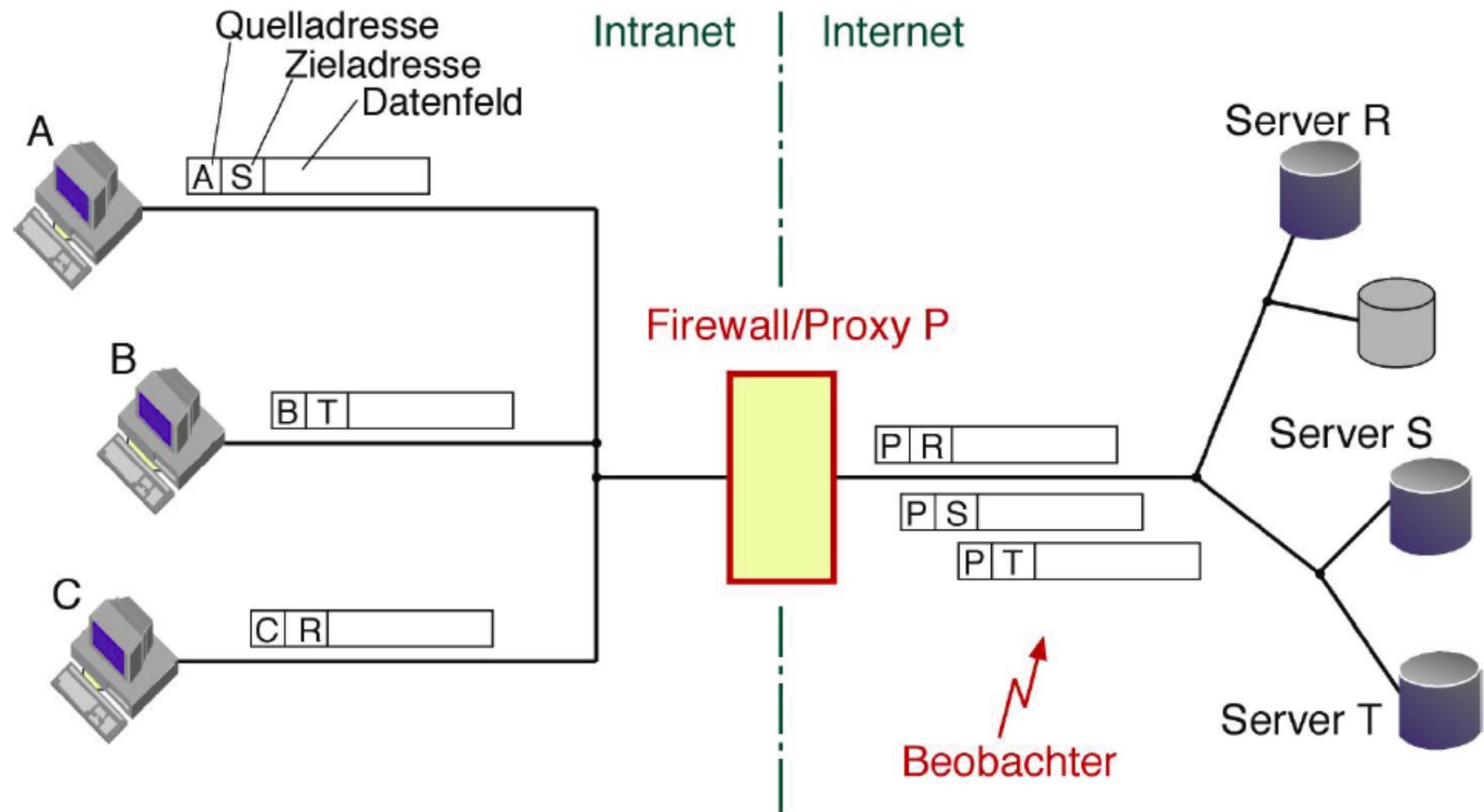
# BEWERTUNG APPLICATION LEVEL GATEWAYS

- Feingranulare, dienstspezifische Filterung
  - Umfangreiche Logging-Möglichkeit und damit Accounting
  - Zustandsbehaftet
  - Inhaltsanalyse (damit z.B. Filterung aktiver Inhalte möglich)
  - Benutzerauthentifizierung und benutzerabhängige Filterung
  - Entkopplung von internem und externem Netz
  - Möglichkeit der Erstellung von Nutzungsprofilen
- 
- Möglichkeit der Erstellung von Nutzungsprofilen
  - Jeder Dienst braucht eigenen Proxy
  - Sicherheit der Proxy-Implementierung und -Konfiguration?
  - Problem von Protokollschwächen bleibt weitgehend bestehen

74

# PROXIES

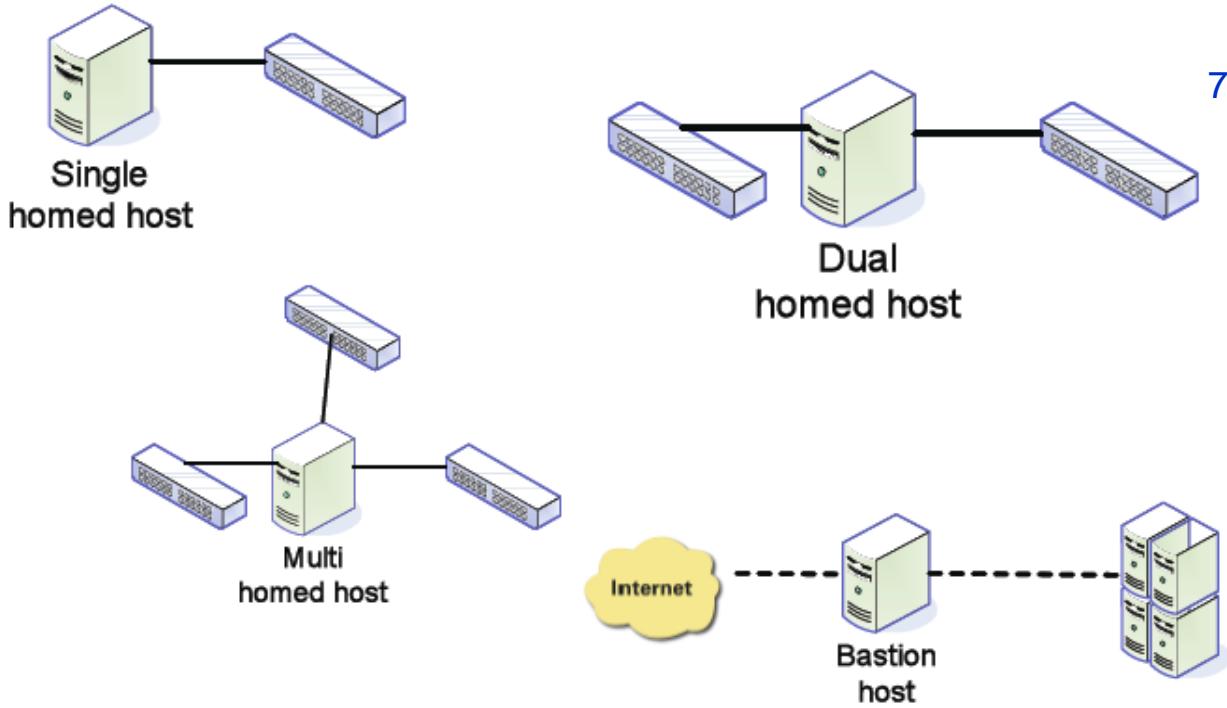
- Schutz vor Beobachtung im Internet



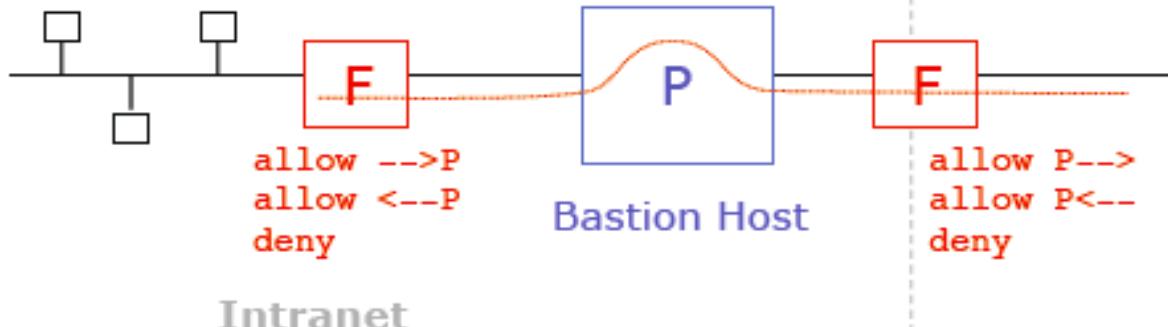
# FIREWALL-ARCHITEKTUREN

- Kombinationen von Firewall-Komponenten und deren Anordnung wird als Firewall-Architektur bezeichnet
- Unterschiedlicher Schutzbedarf führt zur Bildung von Schutzzonen, z.B. öffentlich zugänglich, Mitarbeiternetz, interne Serversysteme, Verwaltungsnetz (Personaldaten), Testnetz, ...

- Bastion Host:  
alle Verbindungen  
zwischen intern und  
extern ausschließlich  
über diesen Host.  
⇒ Host dient an  
forderster Front  
als eine **Bastion** zur  
Angriffsabwehr

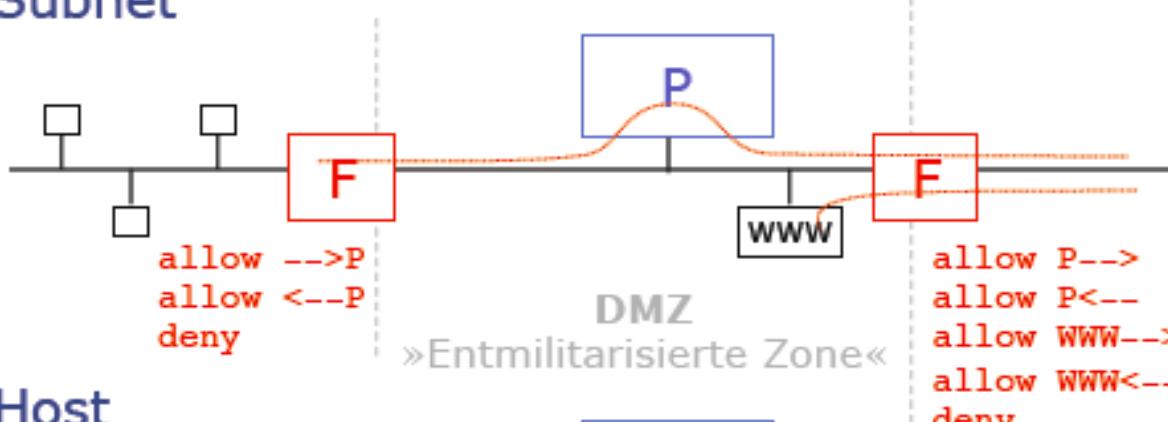


## Dual Homed Gateway



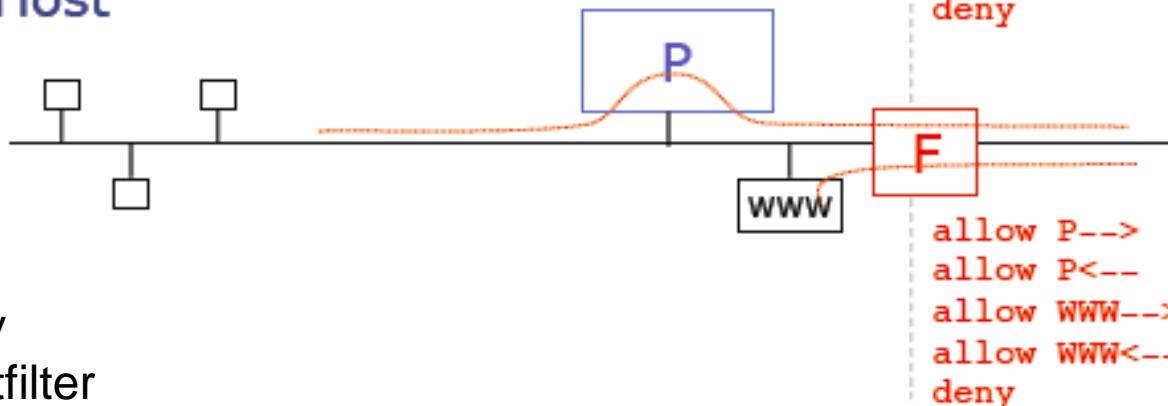
Proxy benötigt 2 Network-Interfaces;  
unflexibel, dafür sehr sicher, da direkte Kommunikation zwischen Inter- und Intranet unmöglich,  
Paketfilter sind theoretisch entbehrlich

## Screened Subnet



weit verbreitet, erreicht annähernd Sicherheit des Dual Homed Gateways, Aufstellen von Internet-Servern innerhalb der DMZ;  
flexibel, da direkte Kommunikation zwischen Inter- und Intranet bei Bedarf möglich

## Screened Host



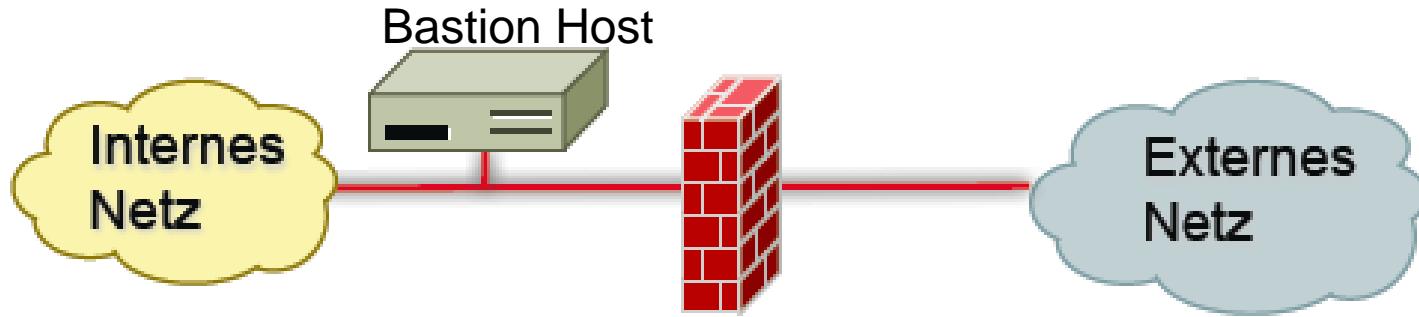
wie Screened Subnet,  
aber kein Schutz vor ausgehenden Paketen mit gespoofter Quell-IP-Adresse

P – Proxy

F – Paketfilter

# SCREENED HOST

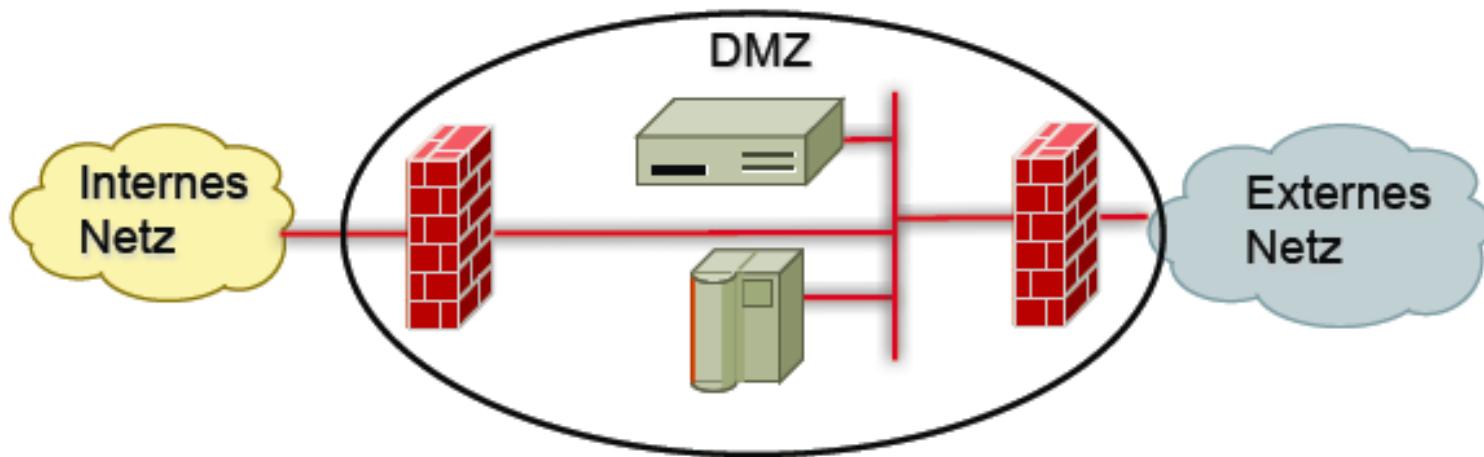
- Firewall (Bastion Host) liegt im internen Netz (nur 1 Interface)
- Verkehr von außen wird über Screening Router/Paketfilter (vor-) gefiltert und i.d.R. zum Bastion Host geleitet
- Bastion Host kann Application Level Gateway oder Circuit Level Gateway realisieren



- Trennung von Paket- und Applikationsfilter
- Vorfilterung des externen Verkehrs
- Hohe Flexibilität
- Pakete können immer noch direkt in internes Netz gelangen

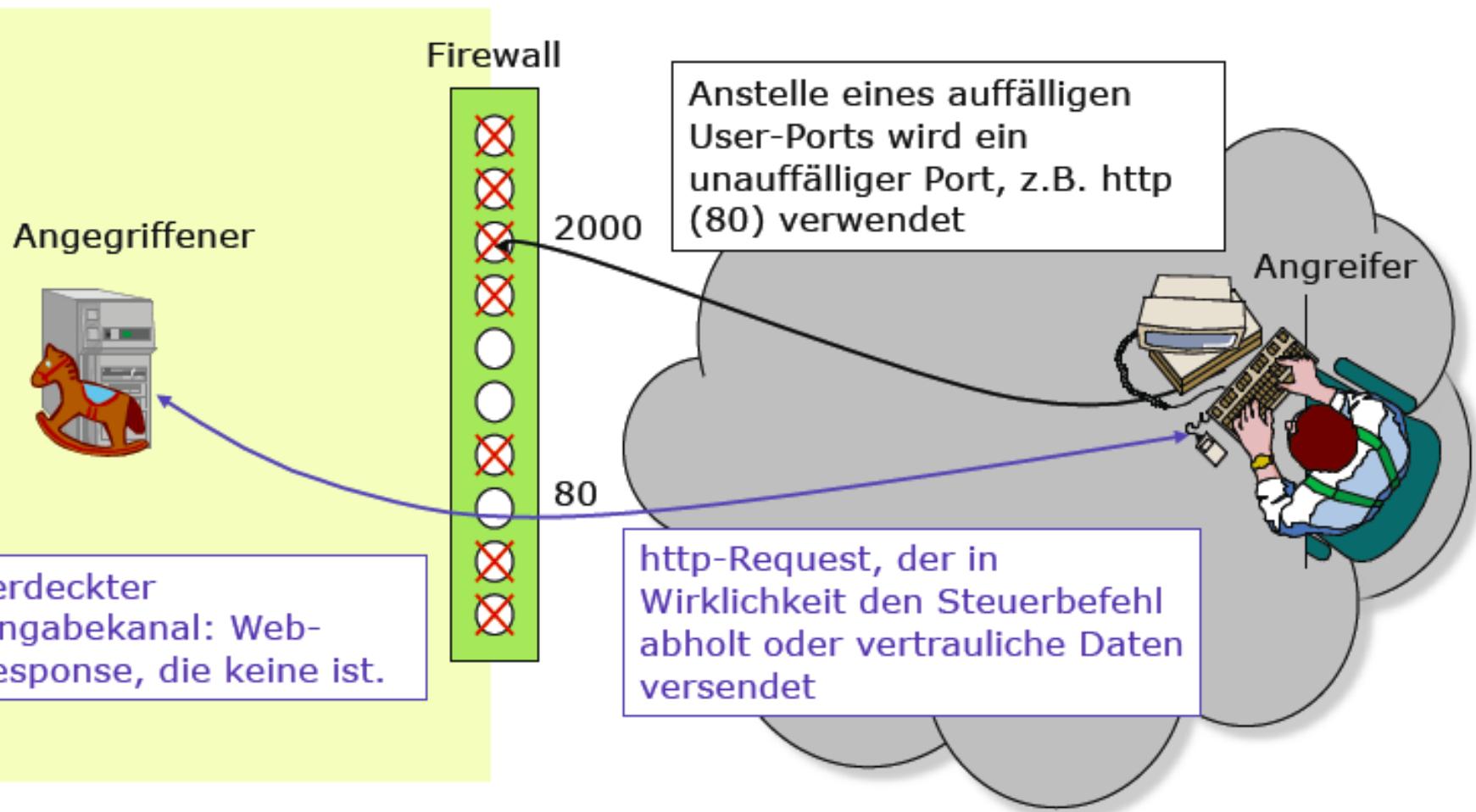
# SCREENED SUBNET

- FW Komponenten liegen in einem eigenen Subnetz (Perimeter Subnet), auch demilitarisierte Zone (DMZ) genannt
- Schutz der DMZ sowohl nach innen als nach außen durch Paketfilter
- Erweiterung der DMZ um dezierte Server, z.B. HTTP/SMTP



- Keine direkte Verbindung von außen nach innen mehr möglich
- Zusätzlicher Grad an Sicherheit
- Interner Router/FW schützt vor Internet und ggf. vor DMZ

# UNTERTUNNELN EINER FIREWALL DURCH TROJANISCHE PFERDE



# GRENZEN VON FIREWALLS

- Firewalls sind eine typische Best-Practice-Technik
- Kompromiss zwischen Schutz und Kosten
- besser aber teurer: jeden einzelnen Rechner im Intranet mit entsprechenden Filterfunktionen ausstatten
- kein Schutz vor Angriffen von innen
- defensive Konfiguration hemmt Produktivität ⇒ Aufweichen der Restriktionen
- selbst bei defensivster Konfiguration ist Überbrücken möglich, Beispiel HTTP-Tunneling
- kein perfekter Schutz vor Viren (verschlüsselte Kommunikation) und überhaupt kein Schutz vor trojanischen Pferden

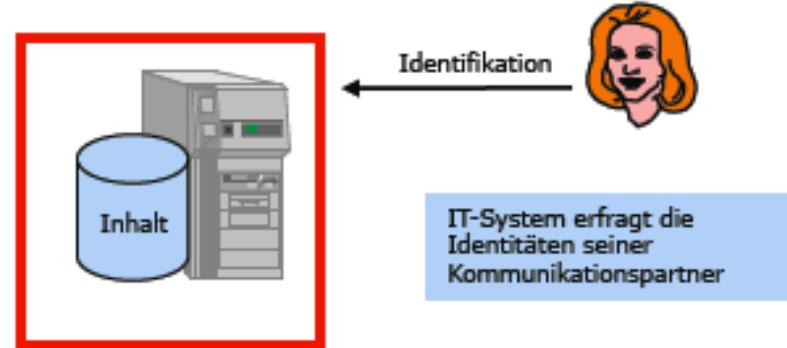
81

# ABGRENZUNG: ZUTRITTS-, ZUGANGS- UND ZUGRIFFSKONTROLLE

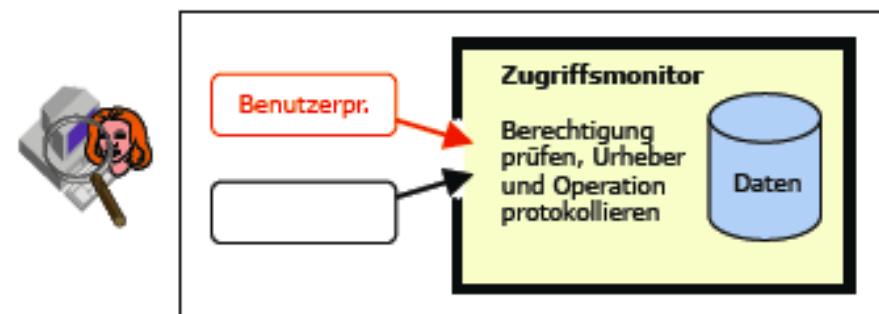
## Zutrittskontrolle



## Zugangskontrolle



## Zugriffskontrolle



## ■ Zugriffskontrollen

- Benutzerbestimmbare Zugriffskontrolle (Discretionary Access Control)
  - Zugriffskontrollmatrix
  - Zugriffskontrolllisten (Access Control Lists)
  - Zugriffsausweise (Capabilities)
- Systembestimmte Zugriffskontrolle (Mandatory Access Control)
  - Bell-La Padula Modell (Vertraulichkeits-Politik)
  - Biba Modell (Integritäts-Politik)
  - Chinese Wall Modell (Interessenkonflikte)
- Zugriffskontrolle in Linux/Unix
- Zugriffskontrolle im Netz
  - Firewall-Typen
  - Firewall-Architekturen

# IT-SICHERHEIT

## 5. KRYPTOGRAPHIE

Wintersemester 2015/16

Prof. Dr. Michael Meier  
Informatik IV  
Arbeitsgruppe IT-Sicherheit



Universität Bonn  
E-Mail: mm@cs.uni-bonn.de

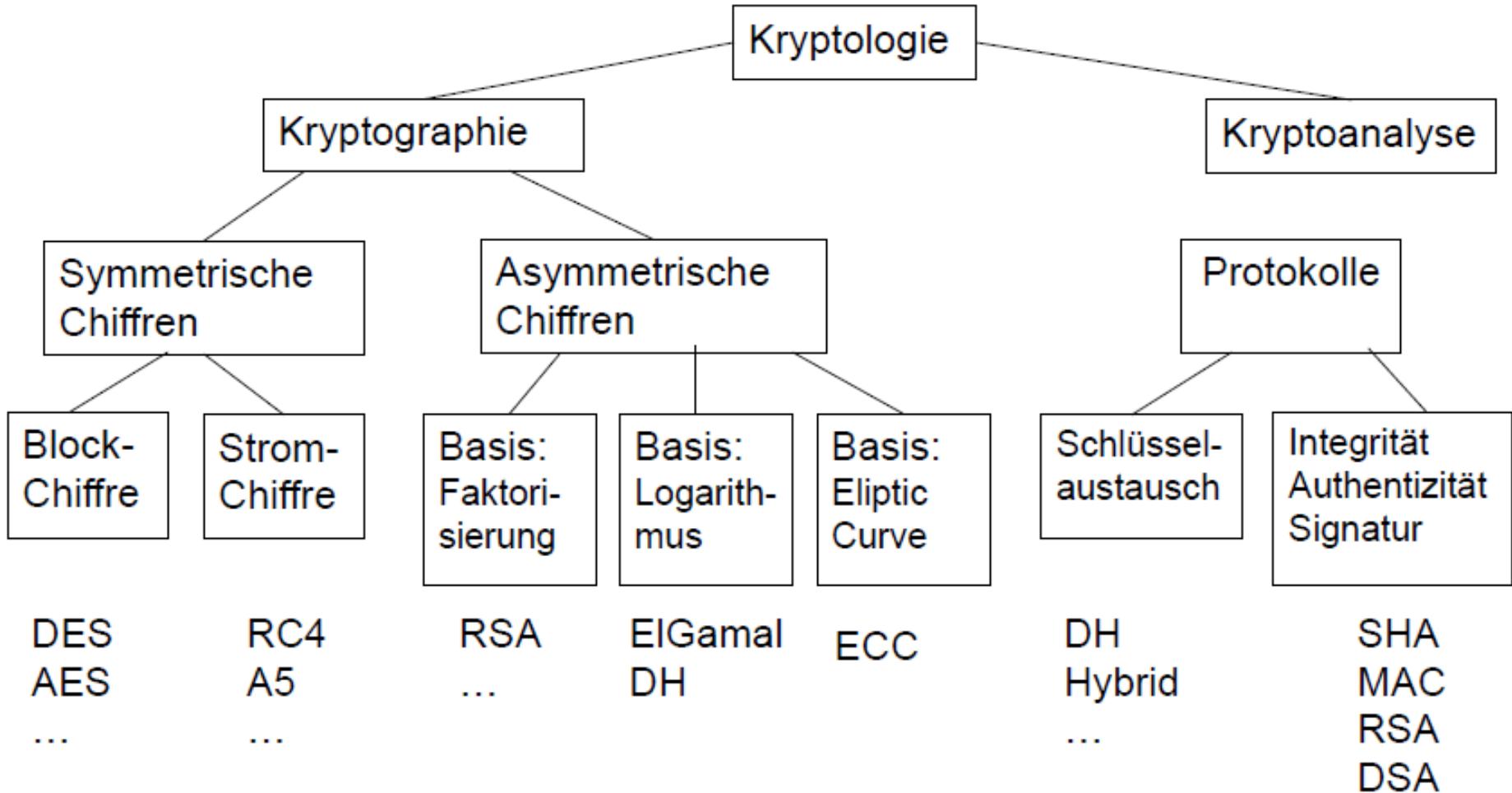
- **Kryptographie:** Lehre von Methoden zur Ver- und Entschlüsselung von Nachrichten zum Zweck der Geheimhaltung von Informationen gegenüber Dritten (Angreifern)
- **Kryptoanalyse:** Wissenschaft von Methoden zur Entschlüsselung von Nachrichten, ohne Zugriff auf den verwendeten Schlüssel zu haben
- **Kryptologie** = Kryptographie + Kryptoanalyse
- **Steganographie:** im Unterschied zur Kryptographie zielen steganographische Methoden darauf bereits die Existenz einer Nachricht zu verbergen (und nicht nur deren Inhalt zu verschleiern)

# KRITERIEN ZUR EINTEILUNG VON KRYPTOSYSTEMEN

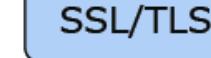
- Zweck bzw. Kryptographische Basisbausteine
  - Verschlüsselungssysteme (auch Konzelationssysteme)
  - Authentifikationssysteme
  - Hashfunktionen
  - Pseudozufallszahlengeneratoren
- Schlüsselbeziehung Sender–Empfänger
  - Symmetrische Systeme
  - Asymmetrische Systeme
- Alphabet, auf dem die Chiffre operiert
  - Blockchiffre: Operiert auf Blöcken von Zeichen
  - Stromchiffren: Operiert auf einzelnen Zeichen
- Längentreue
- Erreichbare Sicherheit

# DIAGRAMM

4



# ANWENDUNGSFALL / SCHLÜSSELBEZIEHUNG

	Konzession (Verschlüsselung)	Authentikation
symmetrische	<i>One-time-pad, DES, Triple-DES, AES, IDEA, A5/1 (GSM), A5/2 (GSM) ...</i>     	<i>Symmetrische Authentikationscodes, CCM, A3 (GSM), ...</i>     
asymmetrische	<i>RSA, ElGamal, McEliece, ...</i>    	<i>RSA, ElGamal, DSA, GMR, ...</i>    

*Algorithmus*

Anwendung

# ERREICHBARE SICHERHEIT

## ■ Sicherheit

- (informations)theoretisch sicher
- kryptographisch stark (beweisbar)
  - gegen aktive Angriffe
  - gegen passive Angriffe
- wohluntersucht (praktisch sicher)
  - Chaos
  - Zahlentheorie
- geheim gehaltene

Komplexitäts-theoretisch sicher

## ■ Kerckhoffs-Prinzip

- Die Sicherheit eines kryptographischen Verfahrens soll von der Geheimhaltung des kryptographischen Schlüssels abhängen.
- Geht zurück auf  
Auguste Kerckhoff: La Cryptographie militaire, 1883

# ANGRIFFSARTEN

- Ciphertext Only Attack
  - Angreifer kennt nur Schlüsseltext
- Known Plaintext Attack
  - Angreifer kenn Klartext-Schlüsseltext-Paare
- [Adaptively] Chosen Plaintext (Ciphertext) Attack
  - Adaptively:
    - Angreifer kann in Abhängigkeit vorheriger gewählter Nachrichten neue Nachrichten wählen
  - Non-adaptively:
    - Angreifer muss alle Nachrichten zu Beginn wählen, kann also nicht abhängig vom Verschlüsselungsergebnis, weitere Nachrichten wählen

# ANGRIFFSARTEN

Authentifikations-  
systeme

Verschlüsselungs-  
systeme

■ Brechen = Fälschen | Entschlüsseln

- Vollständiges Brechen: Finden des Schlüssels
- Universelles Brechen: Finden eines zum Schlüssel äquivalenten Verfahrens
- Nachrichtenbezogenes Brechen: Brechen für einzelne Nachrichten, ohne den Schlüssel selbst in Erfahrung zu bringen
  - selektives Brechen: für eine bestimmte Nachricht
  - existenzielles Brechen: für irgendeine Nachricht

■ Aufwand/Kosten

- Einmalige Kosten, jeder Schlüssel effizient „knackbar“
- Jeder Angriff verursacht Kosten beim Angreifer

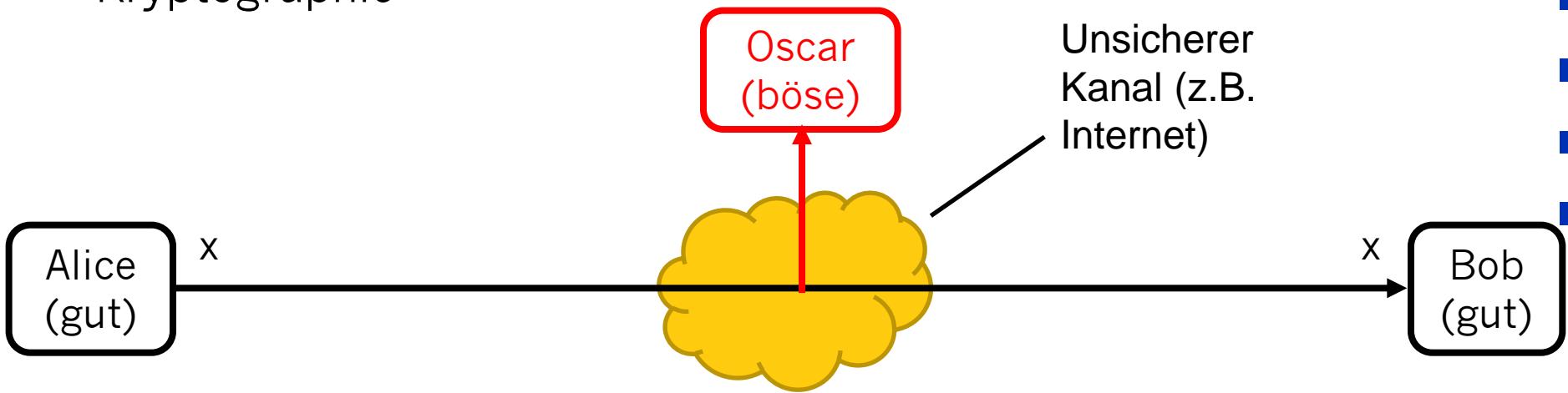
# BASISFAKTA

- **Antike Kryptographie:** frühe Anzeichen für Verschlüsselung in Ägypten ca. 2000 v.u.Z.  
Buchstabenbasierte Verschlüsselungsschemata (z.B. Caesar Chiffre) seither populär
- **Symmetrische Chiffren:** Alle Verschlüsselungsschemata von der Antike bis 1976 waren symmetrische Verfahren.
- **Asymmetrische Chiffren:** 1976 wurde Public-Key (oder asymmetrische) Kryptographie vorgeschlagen von Diffie, Hellman und Merkle.
- **Hybride Schemata:** Die Mehrheit der heutigen Kryptoverfahren sind hybride Systeme, das heißt sie benutzen beides
  - symmetrische Chiffren (z.B. zur Verschlüsselung und Nachrichtenauthentifikation) und
  - asymmetrische Chiffren (z.B. für Schlüsselaustausch und Digitale Signatur).

9

# SYMMETRISCHE KRYPTOGRAPHIE

- Alternative Bezeichnungen: private-key, single-key oder secret-key Kryptographie

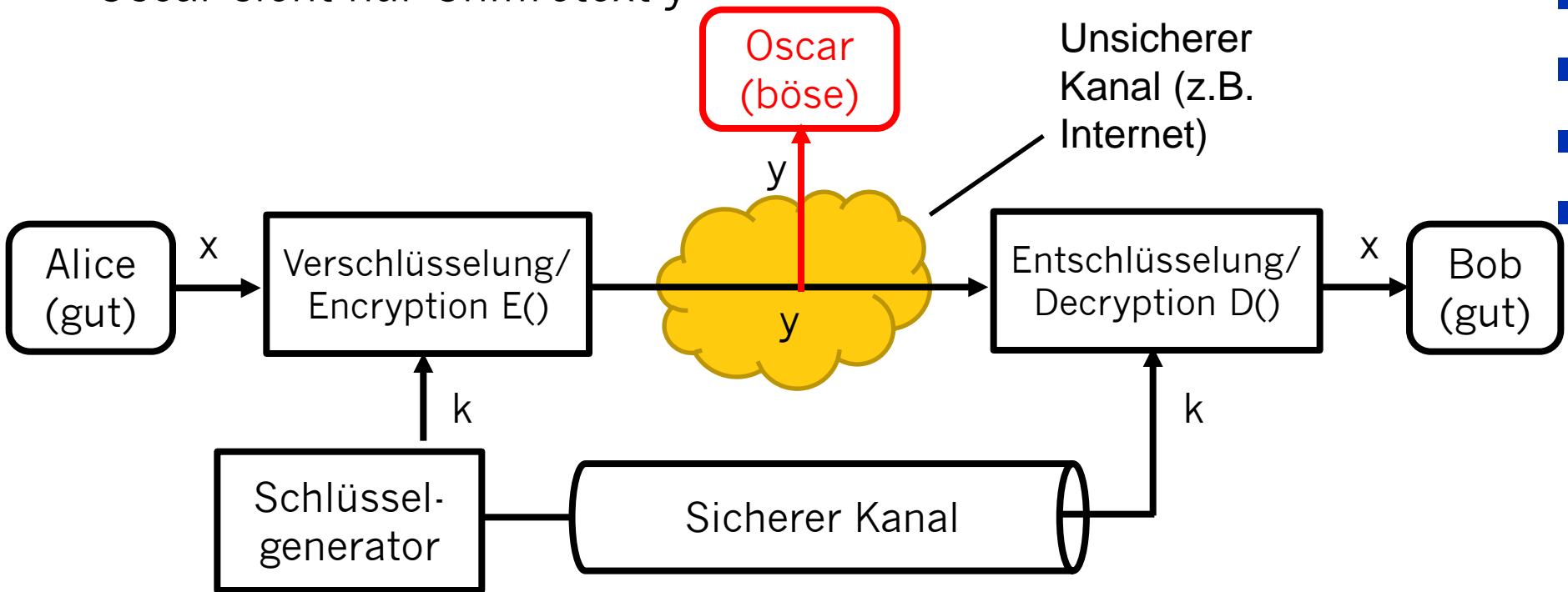


- Problembeschreibung
  - Alice und Bob möchten über einen unsicheren Kanal (z.B. WLAN oder Internet) vertraulich kommunizieren
  - Eine bösartige dritte Partei Oscar hat Zugang zum Kanal, soll aber nicht in der Lage sein, die Kommunikation zu verstehen

# SYMMETRISCHE VERSCHLÜSSELUNG

- Lösung: symmetrische Verschlüsselung

Oscar sieht nur Chiffretext  $y$



- $x$  ist Klartext (plaintext)
- $y$  ist Chiffretext (ciphertext)
- $k$  ist Schlüssel (key)

# SYMMETRISCHE VERSCHLÜSSELUNG

- Verschlüsselungsgleichung:  $y = e_K(x)$
  - Entschlüsselungsgleichung:  $x = d_K(y)$
  - Verschlüsselung und Entschlüsselung sind inverse Operationen wenn derselbe Schlüssel  $k$  auf beiden Seiten verwendet wird
$$d_K(y) = d_K(e_K(x)) = x$$
- 
- Der Schlüssel muss zwischen Alice und Bob über einen sicheren Kanal übertragen werden.
  - Der sichere Kanal kann z.B. realisiert werden durch einen manuelle Übertragung oder einen menschlichen Kurier
  - Das System ist nur sicher, wenn der Angreifer keine Kenntnis von Schlüssel  $k$  erhält.
- 
- ⇒ Das Problem der sicheren Kommunikation wird reduziert auf die sichere Übertragung des Schlüssels  $k$

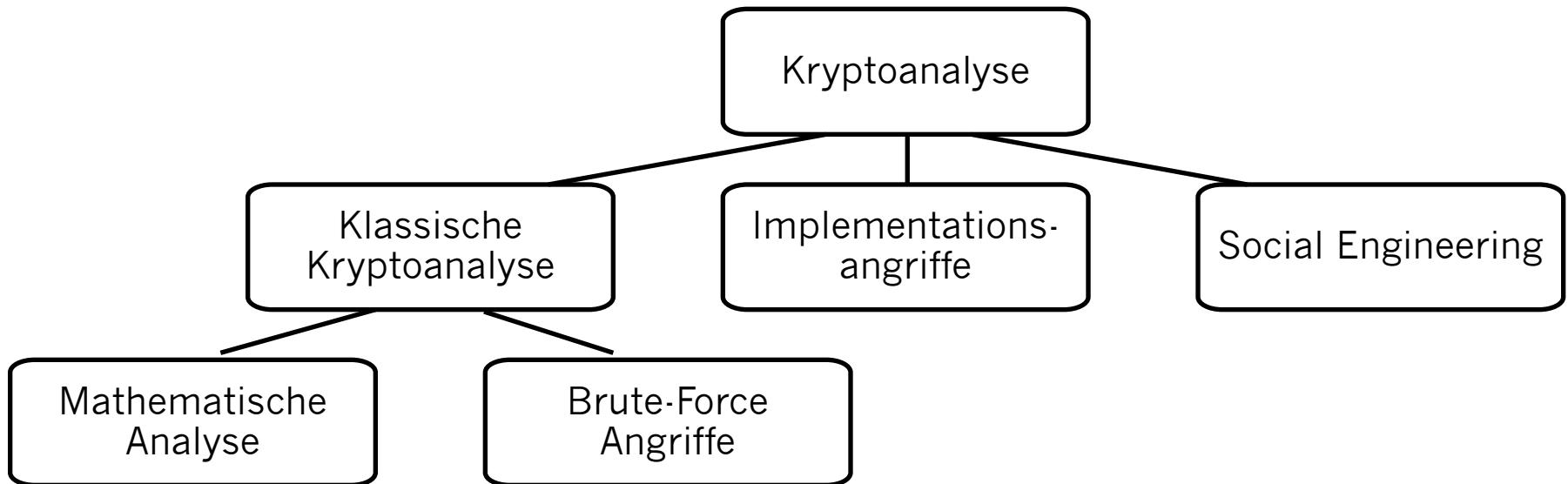
12

# KRYPTOANALYSE

- Warum beschäftigen wir uns mit Kryptoanalyse?
  - Es gibt keinen mathematischen Beweis der Sicherheit von praktikablen Verschlüsselungssystemen.
  - Einziger Weg Zuversicht zu erlangen, dass ein Verschlüsselungssystem sicher ist, ist der Versuch es zu brechen (und dabei zu scheitern)!
- Erreichung von Kerckhoff's Prinzip in der Praxis
  - Ausschließlich Verschlüsselungssysteme verwenden, die weit bekannt sind und von guten Kryptologen mehrere Jahre einer Kryptoanalyse unterzogen wurden!
- Man mag in Versuchung kommen anzunehmen dass Verschlüsselungssysteme sicherer sind wenn ihre Details geheim gehalten werden. Jedoch zeigt die Geschichte wieder und wieder, dass geheime Verschlüsselungssysteme fast immer gebrochen werden können, wenn sie einmal Reverse Engineered wurden.
  - Z.B. Content Scrambling System (CSS) für DVD-Videoinhalte

14

# KRYPTOANALYSE



- **Klassische Kryptoanalyse**
  - Mathematische Analyse
  - Brute-Force Angriffe
- **Implementationsangriffe:** Versuche den Schlüssel durch Reverse Engineering zu extrahieren oder Seitenkanäle wie Stromverbrauch bei Smart Cards
- **Social Engineering:** dem Benutzer durch Austricksen das Passwort entlocken

# BRUTE-FORCE-ANGRIFFE (ODER EXHAUSTIVE KEY SEARCH)

- Betrachtet Chiffre als Black Box
  - Benötigt (mindestens) ein Klartext-Chiffertext-Paar  $(x_0, y_0)$
  - Prüft alle möglichen Schlüssel bis Bedingung erfüllt ist:  
 $d_K(y_0) = x_0$

Schlüssellänge in Bit	Schlüsselraum	Security Life Time (falls Brute-Force der bestmögliche Angriff ist)
64	$2^{64}$	Kurze Zeit (ein paar Tage oder weniger)
128	$2^{128}$	Lange Zeit (Jahrzehnte falls Quantencomputer ausbleiben)
256	$2^{256}$	Lange Zeit (auch resistent gegen Quantencomputer)

- Anmerkung: Quantencomputer existieren aktuell und nicht, und werden möglicherweise niemals existieren

# SUBSTITUTIONS-CHIFFRE

- Historische Verschlüsselung
- Hilfreich zum Verständnis von Brute-Force vs. Analytische Angriffe
- Verschlüsselt Buchstaben statt Bits (wie alle Chiffren bis zum 2. Weltkrieg)
- Idee: ersetze jeden Klartextbuchstaben mit einem festen anderen Buchstaben

Klartext		Chiffertext
A	⇒	k
B	⇒	d
C	⇒	w
	...	

Z.B. wird ABBA verschlüsselt als kddk

- Beispiel: iq ifcc vqqr fb rdq vfllcq na rdq cfjwhwz hr bnnb hcc hwwhbsqvqbre hwq vhlq
- Wie sicher ist die Substitutions-Chiffre?

# ANGRIFFE GEGEN DIE SUBSTITUTIONS-CHIFFRE

## 1. Angriff: Brute-Force Angriff

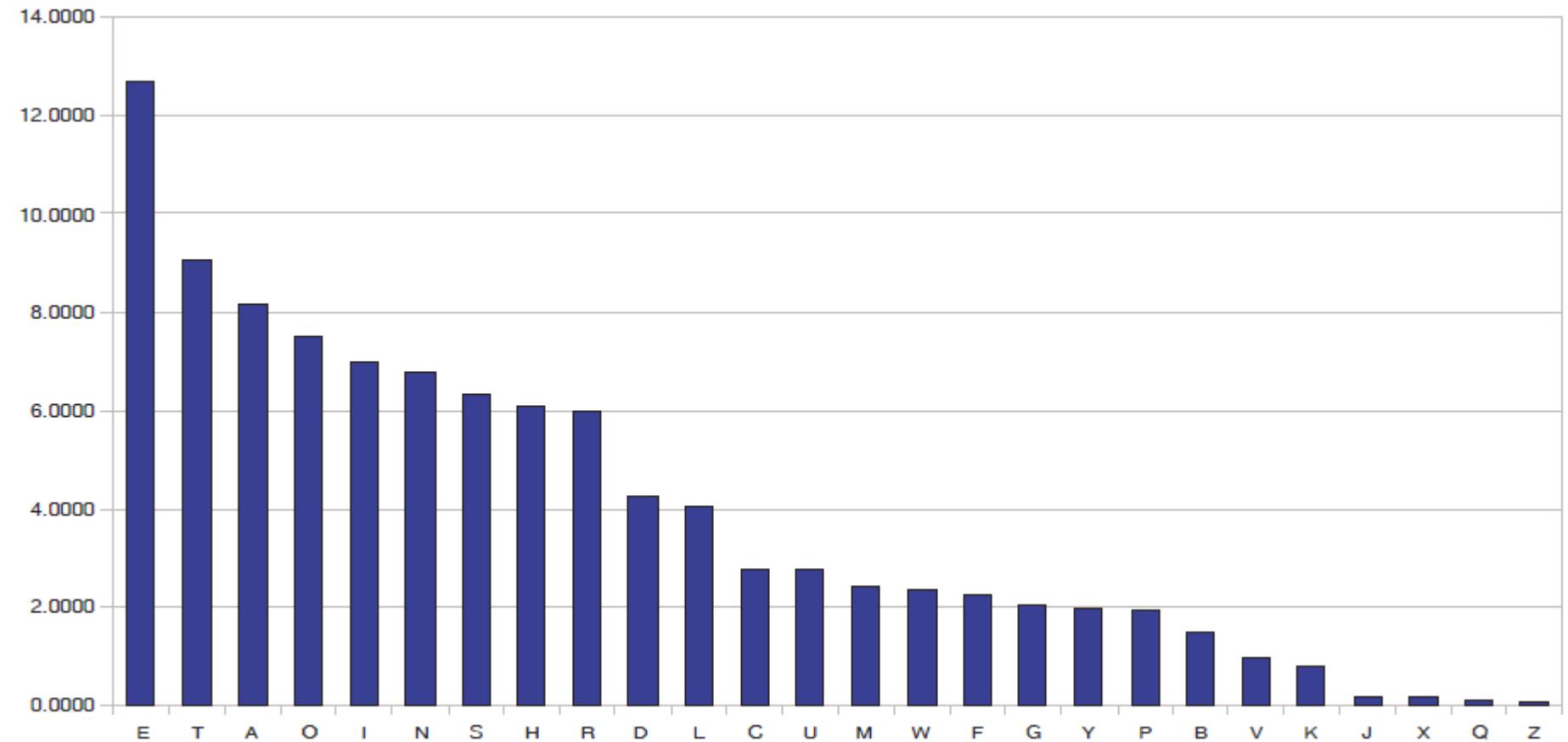
- Probieren aller möglichen Substitutionstabellen bis ein intelligenter Klartext erscheint  
(jede Substitutionstabelle ist ein Schlüssel)
- Wie viele Substitutionstabellen (=Schlüssel) gibt es?
- $26 \times 25 \times \dots \times 3 \times 2 \times 1 = 26! \approx 2^{88}$
- Suche durch  $2^{88}$  Schlüssel ist nicht machbar mit heutigen Computern!  
(siehe frühere Tabelle zu Schlüssellängen)
  
- Können wir jetzt schlussfolgern, dass die Substitutions-Chiffre sicher ist, da Brute Force nicht möglich ist?
  
- Nein! Wir müssen gegen alle Angriffe schützen ...

18

## 2. Angriff: Buchstabenhäufigkeitsanalyse

- Buchstaben haben sehr verschiedene Häufigkeiten in der Englischen Sprache
- Außerdem: die Häufigkeit von Klartextbuchstaben bleibt im Chiffretext erhalten.
- Z.B. „e“ ist der häufigste Buchstabe in Englisch; fast 13% aller Buchstaben in einem englischen Text sind “e”
- Der zweithäufigste ist “t” mit ca. 9%.

# HÄUFIGKEIT VON BUCHSTABEN (IN %) IN ENGLISCHEN TEXTEN



# BRECHEN DER SUBSTITUTIONS-CHIFFRE MIT BUCHSTABENHÄUFIGKEITEN

- Zurück zum Beispiel und häufigsten Buchstaben identifizieren:  
iq ifcc vqqqr fb rdq vfllcq na rdq cfjwhwz hr bnnb hcc  
hwwhbsqvqbvre hwq vhlq
- Ersetzen des Chiffretextbuchstabens q mit En:  
iE ifcc vEEr fb rdE vfllcE na rdE cfjwhwz hr bnnb hcc  
hwwhbsEvEbvre hwE vhlE
- Auswertung der Häufigkeit der verbleibenden Buchstaben liefert den Klartext:  
**WE WILL MEET IN THE MIDDLE OF THE LIBRARY AT NOON ALL ARRANGEMENTS ARE MADE**

21

# BRECHEN DER SUBSTITUTIONS-CHIFFRE MIT BUCHSTABENHÄUFIGKEITEN

- Nicht nur die Häufigkeit von einzelnen Buchstaben kann für einen Angriff genutzt werden sondern auch die Häufigkeit von Buchstabenpaaren (z.B. „th“ kommt häufig vor in englischen Texten), Buchstaben-Triple, etc.
- **Wichtige Lektion:**  
Obwohl die Substitutionschiffre einen ausreichend großen Schlüsselraum (ca.  $2^{88}$ ) nutzt, kann sie mit analytischen Methoden gebrochen werden.  
⇒ Ein Verschlüsselungsverfahren muss **allen** Arten von Angriffen widerstehen.

22

# CAESAR (ODER VERSCHIEBE) CHIFFRE

- Antikes Verschlüsselungsverfahren, angeblich genutzt von Julius Caesar
- Ersetzt jeden Klartextbuchstaben durch einen anderen
- Einfache Ersetzungsregel: nimm den Buchstaben der im Alphabet  $k$  Positionen weiter hinten steht
- Erfordert Abbildung von Buchstaben auf Zahlen:

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

- Beispiel für  $k = 7$   
Klartext = ATTACK = 0, 19, 19, 0, 2, 10  
Chiffretext = haahjr = 7, 0, 0, 7, 9, 17
- Buchstaben brechen am Ende des Alphabets um  
mathematisch: Reduktion modulo 26, z.B.  $19 + 7 = 26 \equiv 0 \pmod{26}$

23

# CAESAR (ODER VERSCHIEBE) CHIFFRE

- Elegante mathematische Formulierung der Chiffre

Seien  $k, x, y \in \{0, 1, \dots, 25\}$

Verschlüsselung:  $y = e_k(x) \equiv x + k \pmod{26}$

Entschlüsselung:  $x = d_k(y) \equiv y - k \pmod{26}$

- Ist die Verschiebechiffre sicher?
- Nein, verschiedene Angriffe möglich, einschließlich
  - Brute force (Schlüsselraum nur 26)
  - Buchstabenhäufigkeitsanalyse

# SPALTEN-TRANSPOSITION (SKYTALA)

## ■ Skytala

- Ca. 2400 Jahre alte griechische Chiffre
- Zylinder mit gewickeltem Papierstreifen, schreiben, abwickeln
- Empfänger hat Zylinder mit gleichem Durchmesser



Bild: Wikipedia (CC BY-SA 3.0)

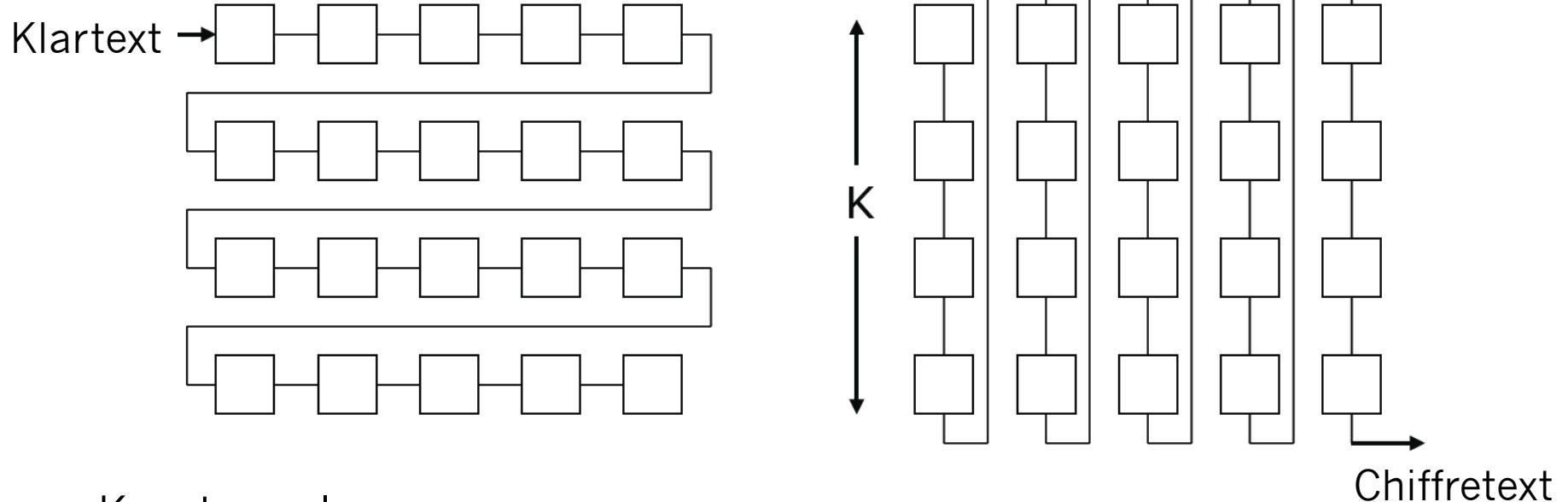
## ■ Kryptoanalyse

- Heute: Durchprobieren (sehr kleiner Schlüsselraum)
- Statistische Analyse (Bigramme)

# SPALTEN-TRANSPOSITION (SKYTALA)

## ■ Skytala

- Ca. 2400 Jahre alte griechische Chiffre
- Zylinder mit gewickeltem Papierstreifen, schreiben, abwickeln
- Empfänger hat Zylinder mit gleichem Durchmesser



## ■ Kryptoanalyse

- Heute: Durchprobieren (sehr kleiner Schlüsselraum)
- Statistische Analyse (Bigramme)

# SKYTALE: STATISTISCHE ANALYSE

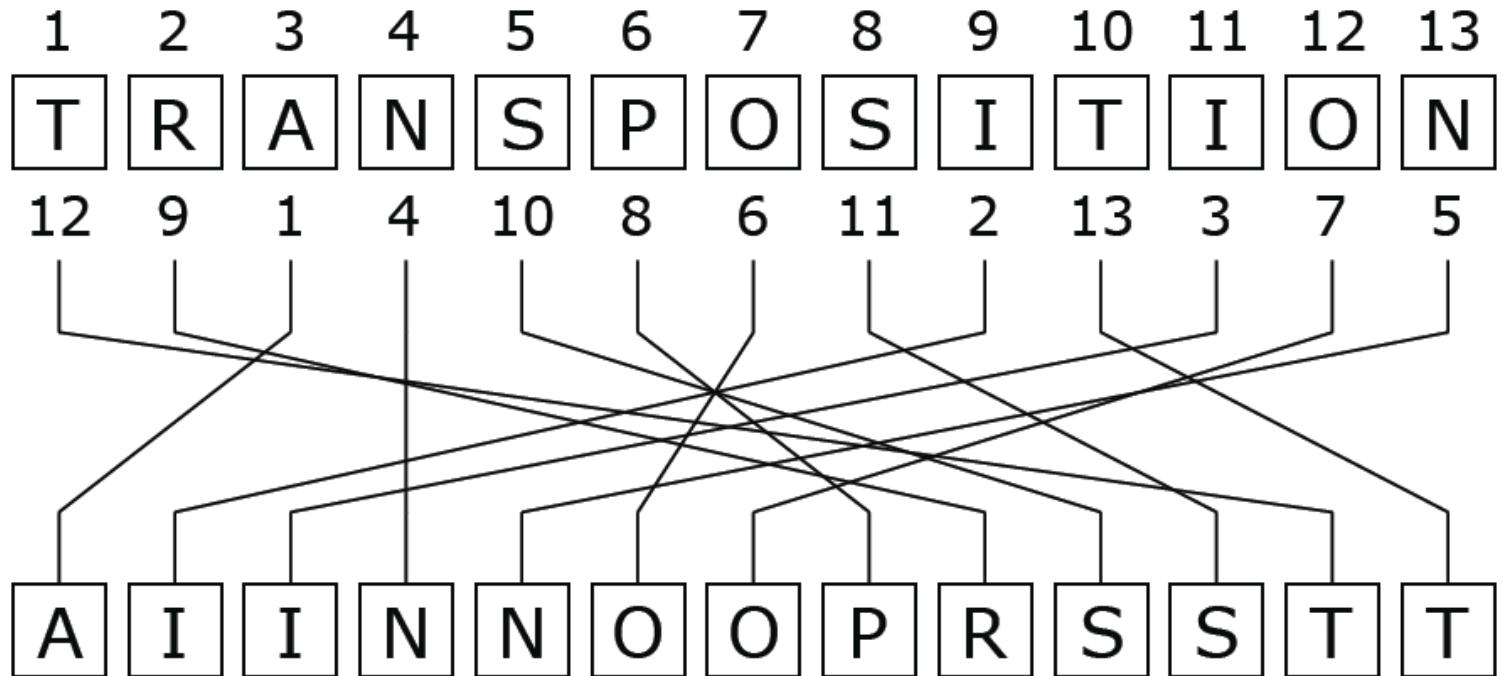
- Vorgehen: Suche typische Bigramme (z.B. EN, ER, CH, ...) und ermittle Häufigkeit und Buchstabenabstände.
- Beispiel
  - Klartext: VERSCHLUESSELNMACHTGROSSENSPASS
  - $k = 5$   
VERSCHL  
UESSELN  
MACHTGR  
OSSENSP  
ASSXYZX
  - Chiffretext: VUMOAEEASSRSCSSSSHEXCETNYHLGSZLNRPX

Bigramm	Chiffretext	Abstand/Häufigk.
EN:	VUMOAEEASSRSCSSSSHEXCETNYHLGSZLNRPX:	2/1, 5/1, ...
ER:	VUMOAEEASSRSCSSSSHEXCETNYHLGSZLNRPX:	5/1, 4/1, ...
EI:	VUMOAEEASSRSCSSSSHEXCETNYHLGSZLNRPX:	0
CH:	VUMOAEEASSRSCSSSSHEXCETNYHLGSZLNRPX:	5/2, ...

- Abstand 5 kommt am Häufigsten vor, teste, ob Text in 5 Zeilen sinnvoll -> gebrochen

# „FREI“ PERMUTATION

- Idee:
  - Zeichen werden nach Vorschrift vertauscht
- Beispiel



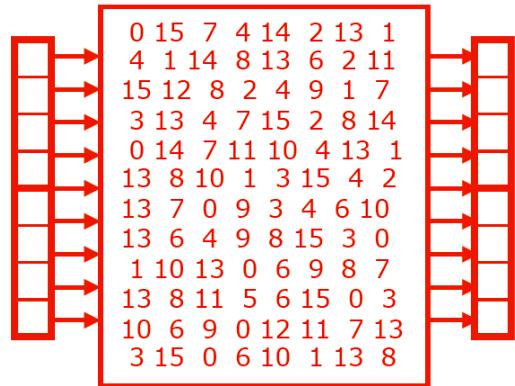
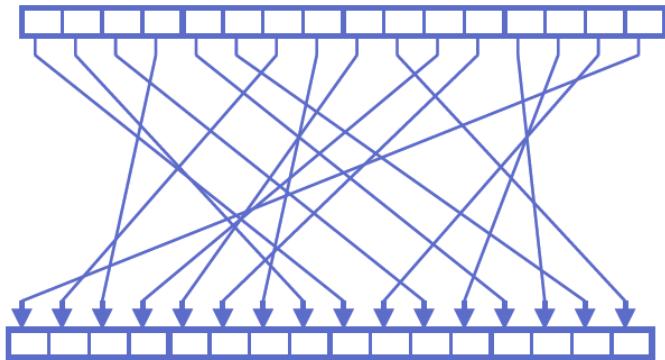
# TRANSPOSITIONEN

- Bewertung
  - Anforderung an den Schlüssel
    - Gutes Mischen der Klartextzeichen
    - Bestenfalls ist der mittlere **Abstand zwischen den Bildpositionen benachbarter Urbilder** etwa die halbe Blockgröße.
  - Erläuterung:
    - **Abstand zwischen den Bildpositionen:** Abstand zwischen den Zeichen im Ciphertext
    - **Benachbarte Urbilder:** benachbarte Klartextzeichen

29

# KLASSISCHE CHIFFREN: SYSTEMATIK

- Transpositionschiffre: (Permutationen)
  - Veränderung der Anordnung von Schriftzeichen
- Substitutionschiffre:
  - Systematische Ersetzung von Schriftzeichen
- Produktchiffre:
  - Kombination von Transpositionen und Substitutionen
  - Vorläufer der modernen symmetrischen Kryptographie, bei der Permutationen und Substitutionen (meist) iterativ angewendet werden



30

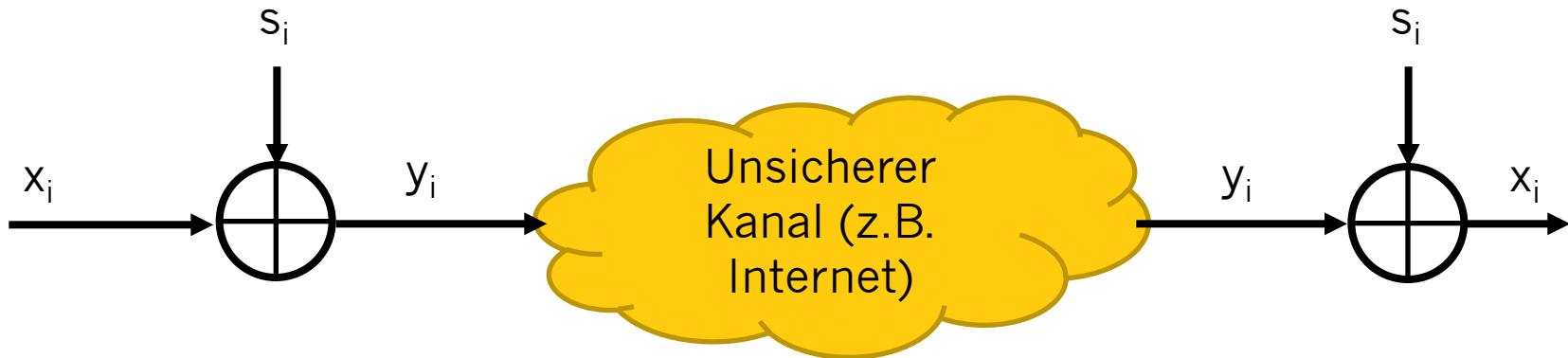
# STROMCHIFFREN VS. BLOCKCHIFFREN

- Stromchiffren
  - verschlüsseln Bits individuell
  - gewöhnlich klein und schnell
    - verbreitet in eingebetteten Geräten (z.B. A5/1 für GSM Telefone)
- Blockchiffren
  - Verschlüsseln immer einen vollen Block (mehrere Bits)
  - Sind üblich für Internetanwendungen

31

# VERSCHLÜSSELUNG UND ENTVERSCHLÜSSELUNG MIT STROMCHIFFREN

- Klartext  $x_i$ , Chiffretext  $y_i$  und Schlüsselstrom  $s_i$  bestehen aus individuellen Bits

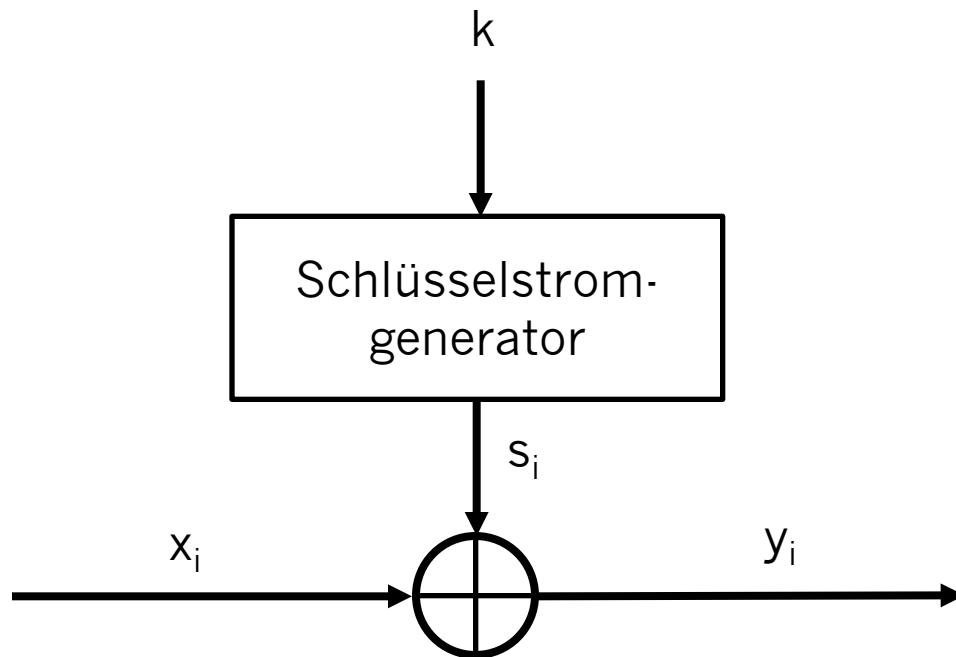


- Verschlüsselung und Entschlüsselung sind einfache Additionen modulo 2 (aka XOR)
- Verschlüsselung und Entschlüsselung sind dieselben Funktionen  
Verschlüsselung:  $y_i = e_{s_i}(x_i) = x_i + s_i \bmod 2 \quad x_i, y_i, s_i \in \{0, 1\}$   
Entschlüsselung:  $x_i = e_{s_i}(y_i) = y_i + s_i \bmod 2$

32

# STROMCHIFFREN (SYNCHRON VS. ASYNCHRON)

- Die Sicherheit von Stromchiffren hängt ausschließlich vom Schlüsselstrom  $s_i$  ab
  - sollte zufällig sein, d.h.  $\Pr(s_i=0) = \Pr(s_i=1) = 0,5$
  - muss reproduzierbar sein bei Sender und Empfänger
  - hängt von Schlüssel  $k$  ab



# WARUM IST MODULE 2 ADDITION EINE GUTE VERSCHLÜSSELUNGSFUNKTION?

- Module 2 Addition ist äquivalent zur XOR Operation
- Bei perfekt zufälligem Schlüsselstrom  $s_i$  hat jedes Chiffretext-Ausgabebit eine 50% Chance 0 oder 1 zu sein
- ⇒ Gute statistische Eigenschaft für Chiffretext
- Invertierung von XOR ist einfach  
da es dieselbe XOR Operation ist

34

$x_i$	$s_i$	$y_i$
0	0	0
0	1	1
1	0	1
1	1	0

# STROMCHIFFRE - DATENDURCHSATZ

35

Cipher	Key length	Mbit/s
DES	56	36.95
3DES	112	13.32
AES	128	51.19
RC4 (stream cipher)	(choosable)	211.34

# ZUFALLSZAHLENGENERATOREN

36

Zufallszahlen-  
generatoren

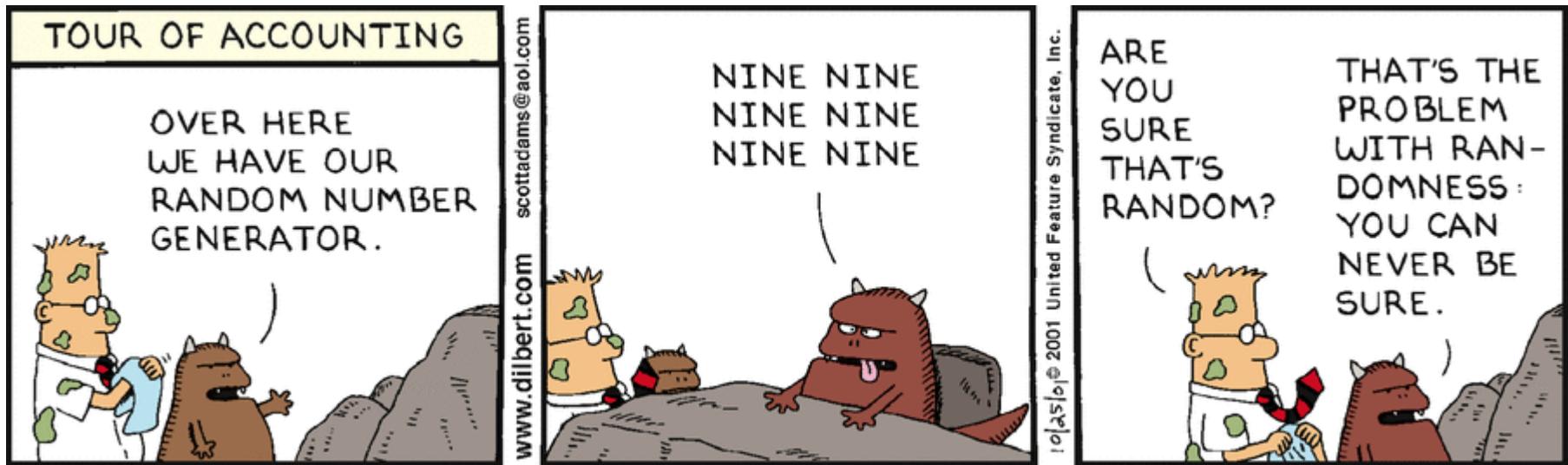
Echte  
Zufallszahlen

Pseudozufalls-  
zahlen-  
generatoren

Kryptographisch  
sichere Pseudo-  
zufallszahlen-  
generatoren

# ECHTE ZUFALLSZAHLGENERATOREN

- True Random Number Generator (TRNG)
- Basieren auf physikalischen Zufallsprozessen: z.B. Münzwurf, Würfel, Halbleiterrauschen, Radioaktiver Zerfall, Maus-Bewegungen
- Ausgabestrom  $s_i$  sollte gute statistische Eigenschaften aufweisen
  - $\Pr(s_i = 0) = \Pr(s_i = 1) = 50\%$  (oft erreicht durch Nachbearbeitung)
- Ausgabe kann weder vorhergesagt noch reproduziert werden
- Typische Einsatzgebiete: Generierung von Schlüsseln, und nonces (Einmalwerte) und vieles weitere



# PSEUDO-ZUFALLSZAHLGENERATOREN

- Pseudo Random Number Generator – PRNG
- Generieren Sequenzen ausgehend/abhängig von initialem Seed-Wert
- Typischerweise besitzt Ausgabestrom gute statistische Eigenschaften
- Ausgabe kann (zu gegebenem Seed) reproduziert und vorhergesagt werden
- Oft rekursiv berechnet:  
 $s_0 = \text{seed}$   
 $s_{i+1} = f(s_i, s_{i-1}, \dots, s_{i-t})$
- Beispiel: rand() Funktion in ANSI C  
 $s_0 = 12345$   
 $s_{i+1} = 1103515245 s_i + 12345 \bmod 2^{31}$
- Die meisten PRNG haben schlechte kryptographische Eigenschaften

38

# KRYPTOANALYSE EINES EINFACHEN PRNG

- Einfacher PRNG

$s_0 = \text{seed}$

$$s_{i+1} = As_i + B \bmod m$$

- Angenommen

Unbekannt sind A, B, und  $s_0$  als Schlüssel

Größe von A, B,  $s_0$  sei 100 Bit

300 Bit Ausgabe sind bekannt, z.B.  $s_1, s_2, s_3$

Lösung von

$$s_2 = As_1 + B \bmod m$$

$$s_3 = As_2 + B \bmod m$$

legt direkt A und B offen und alle  $s_i$  können leicht berechnet werden.

- Schlechte kryptographische Eigenschaften aufgrund der Linearität der meisten PRNGs

39

# KRYPTOGRAPHISCH SICHERE PSEUDO-ZUFALLSZAHLGENERATOREN

- Cryptographically Secure Pseudo Random Number Generator (CSPRNG)
- Spezielle PRNG mit zusätzlicher Eigenschaft
  - Ausgabe darf nicht vorhersagbar sein
- Präziser: Gegeben  $n$  aufeinanderfolgende Bits der Ausgabe  $s_i$ , die folgenden Ausgabebits  $s_{n+1}$  können nicht vorhergesagt werden
- CSPRNG werden benötigt in der Kryptographie, insbesondere für Stromchiffre
- Anmerkung: Es gibt fast keine andere Anwendung die Unvorhersagbarkeit erfordert, aber viele, viele Systeme die PRNGs benötigen.

40

# KRYPTOGRAPHISCH SICHERE PSEUDO-ZUFALLSZAHLGENERATOREN

- Cryptographically Secure Pseudo Random Number Generator (CSPRNG)
- Spezielle PRNG mit zusätzlicher Eigenschaft
  - Ausgabe darf nicht vorhersagbar sein
- Präziser: Gegeben  $n$  aufeinanderfolgende Bits der Ausgabe  $s_i$ , die folgenden Ausgabebits  $s_{n+1}$  können nicht vorhergesagt werden
- CSPRNG werden benötigt in der Kryptographie, insbesondere für Stromchiffre
- Anmerkung: Es gibt fast keine andere Anwendung die Unvorhersagbarkeit erfordert, aber viele, viele Systeme die PRNGs benötigen.

40

# ONE-TIME PAD (OTP) – AUCH VERNAM-CHIFFRE

- Uneingeschränkt/informationstheoretisch sicheres Kryptosystem:
  - Ein Kryptosystem ist uneingeschränkt sicher wenn es mit unbegrenzten Berechnungsressourcen nicht gebrochen werden kann.

## One-Time Pad

- Klartext, Chiffertext und Schlüssel seien individuelle Bits  $x_i, y_i, k_i \in \{0,1\}$ .

41

Verschlüsselung:

$$e_{ki}(x_i) = x_i \oplus k_i$$

Entschlüsselung:

$$d_{ki}(y_i) = y_i \oplus k_i$$

- OTP ist uneingeschränkt sicher wenn und nur wenn der Schlüssel  $k_i$  nur einmal benutzt wird.

# ONE-TIME PAD (OTP)

- Uneingeschränkt/informationstheoretisch sicheres Kryptosystem:
  - $y_0 = x_0 \oplus k_0$
  - $y_1 = x_1 \oplus k_1$
  - :
- Jede dieser Gleichungen ist eine lineare Gleichung mit 2 Unbekannten
  - ⇒ für jedes  $y_i$  sind  $x_i = 0$  und  $x_i = 1$  gleich wahrscheinlich!
  - ⇒ Dies gilt wenn  $k_0, k_1, \dots$  unabhängig sind, d.h. alle  $k_i$  müssen echt zufällig generiert sein
  - ⇒ es kann gezeigt werden, dass dieses System von Gleichungen beweisbar nicht gelöst werden kann
- **Nachteil:** Für fast alle Anwendungen ist OTP **unpraktikabel**, da der Schlüssel genauso lang wie die Nachricht sein muss.

42

# BASISOPERATIONEN VON BLOCKCHIFFREN: KONFUSION UND DIFFUSION

- Es gibt zwei Basisoperationen mit denen starke kryptographische Algorithmen erstellt werden können:
  - **Konfusion:** eine Verschlüsselungsoperation durch welche die Beziehung zwischen Schlüssel und Chiffretext verschleiert (bzw. möglichst komplex gestaltet) wird.  
Typischerweise erreicht durch Substitution.
  - **Diffusion:** eine Verschlüsselungsoperation bei der ein Klartextzeichen möglichst viele Chiffretextzeichen beeinflusst, mit dem Ziel statistische Eigenschaften des Klartextes zu verbergen.  
Kann durch Permutationen erreicht werden.
    - Kleine Änderungen am Klartext bewirken große/unvorhersehbare Änderungen am Chiffretext
    - ⇒ Lawineneffekt (Avalanche-Effekt)
- Beide Operationen für sich können keine Sicherheit bieten.
  - ⇒ Verbindung von Konfusion und Diffusion zur Erstellung so genannter Produktchiffren

43

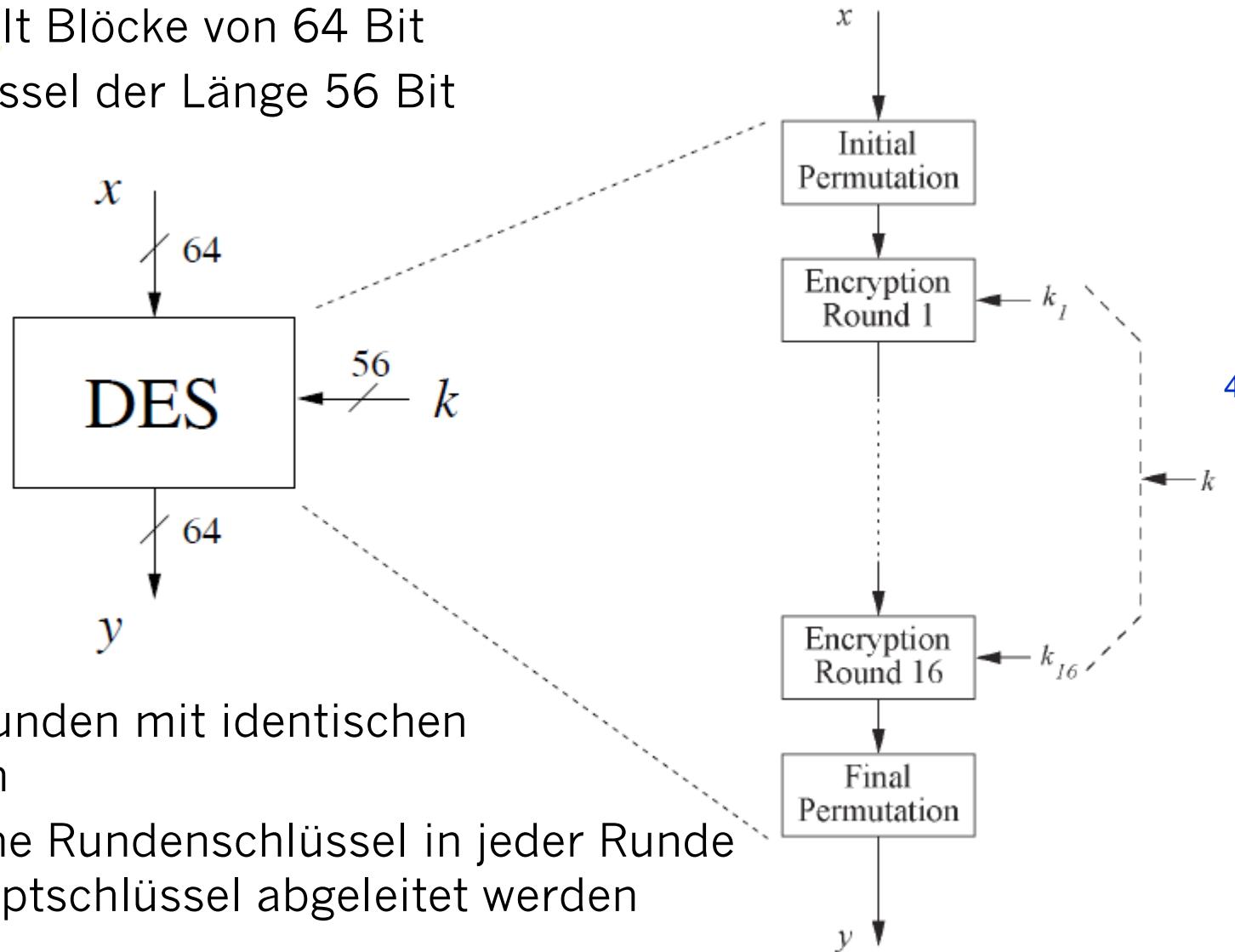
# DATA ENCRYPTION STANDARD (DES)

## ■ Fakten

- Data Encryption Standard (DES) verschlüsselt **Blöcke von 64 bit.**
- entwickelt von IBM basierend auf der Chiffre **Lucifer** unter Einfluss der **National Security Agency (NSA)**, die Entwurfskriterien für DES wurden nicht veröffentlicht
- **Standardisiert 1977** durch das **National Bureau of Standards (NBS)** heute **National Institute of Standards and Technology (NIST)**
- Populärste Blockchiffre in den letzten 30 Jahren.
- Bei weitem am besten untersuchter symmetrischer Algorithmus.
- Heute als unsicher betrachtet aufgrund zu kurzer **Schlüssellänge von 56 bit.**
- Aber **3DES** als sicher angesehen und heute noch weit verbreitet
- In 2000 ersetzt durch den **Advanced Encryption Standard (AES)**

# ÜBERBLICK DES DES-ALGORITHMUS

- Verschlüsselt Blöcke von 64 Bit
- Nutzt Schlüssel der Länge 56 Bit

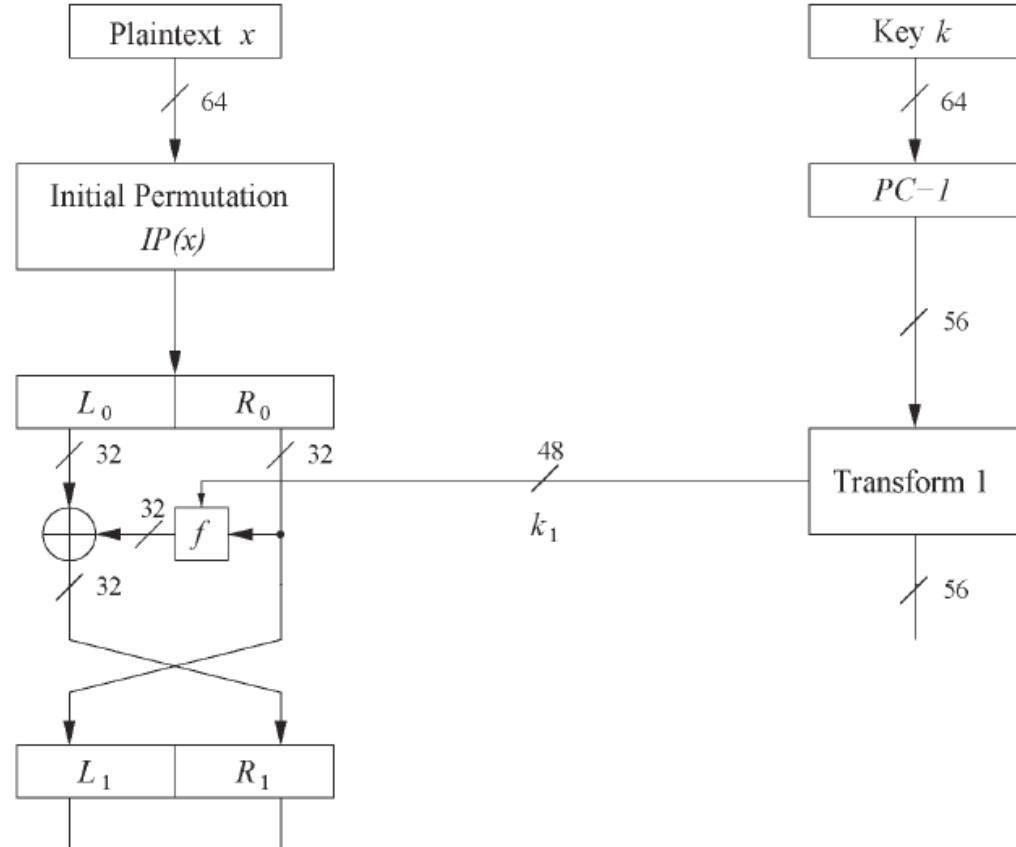


- Nutze 16 Runden mit identischen Operationen
- Verschiedene Rundenschlüssel in jeder Runde die aus Hauptschlüssel abgeleitet werden

# DES FEISTEL-NETZWERK

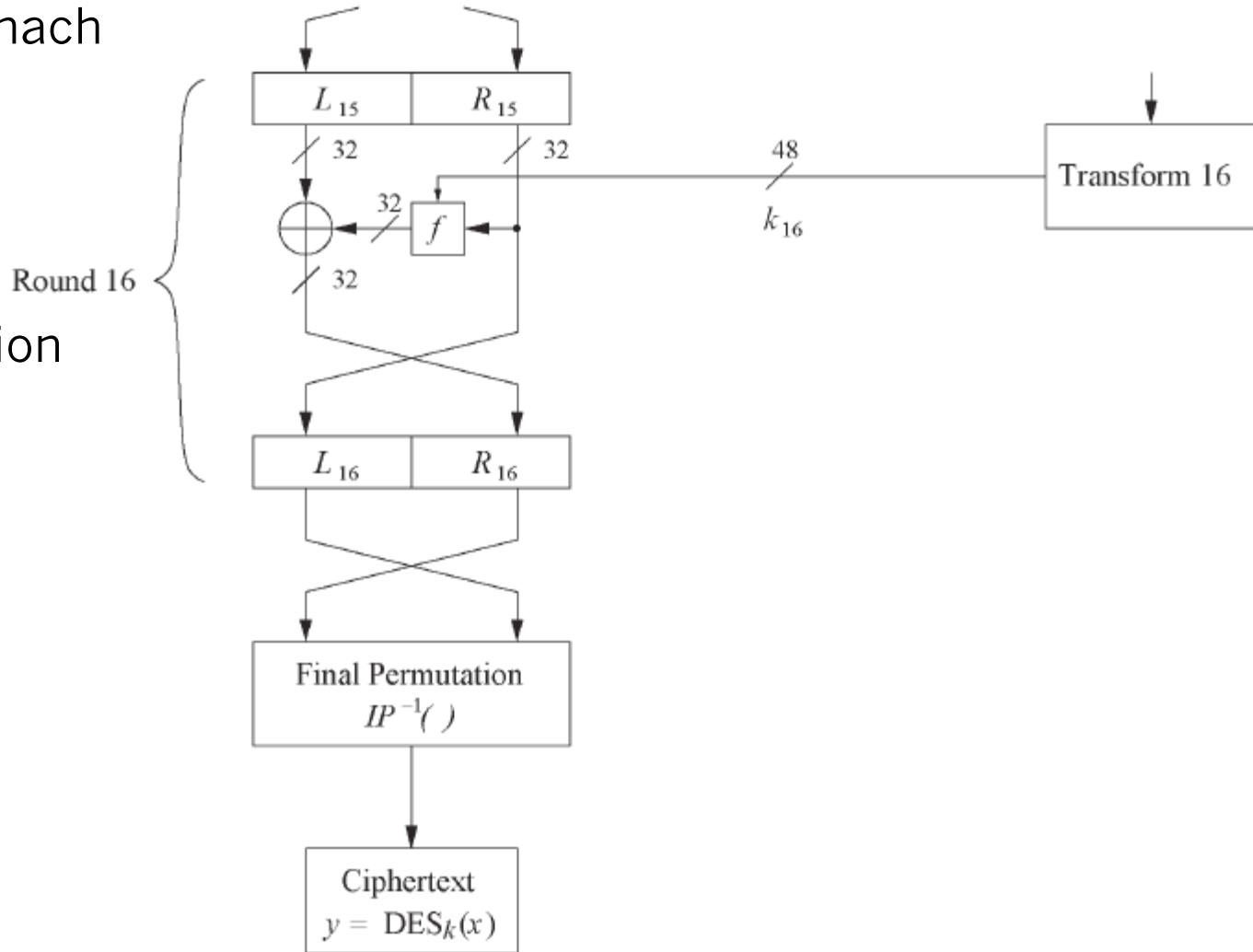
- Grundstruktur von DES ist so genanntes Feistel-Netzwerk
- Vorteil: Ver- und Entschlüsselung unterscheiden sich nur in der Verwendung der Rundenschlüssel
- Bitweise initiale Permutation dann 16 Runden
- 1. Klartext in zwei 32-Bit-Hälften  $L_i$  und  $R_i$
- 2.  $R_i$  an Funktion  $f$  übergeben deren Ergebnis mit  $L_i$  XOR-verknüpft wird
- 3. linke und rechte Hälfte werden vertauscht
- Runde entspricht  $L_i = R_{i-1}$ ,

$$R_i = L_{i-1} \oplus f(R_{i-1}, k_i)$$



# DES FEISTEL-NETZWERK

- L und R werden nach Runde 16 erneut vertauscht
- gefolgt von finaler Permutation

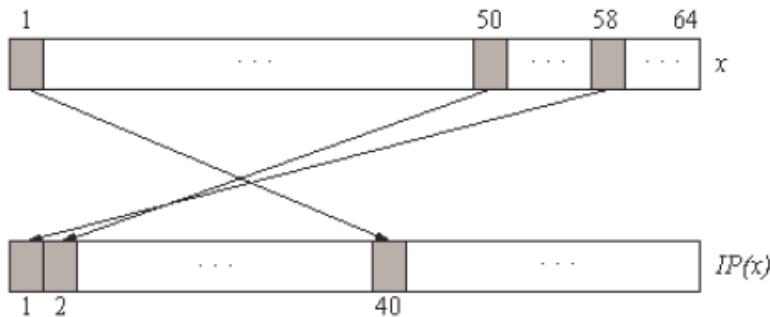


# INNEREIN VON DES

## ■ Initiale und finale Permutation

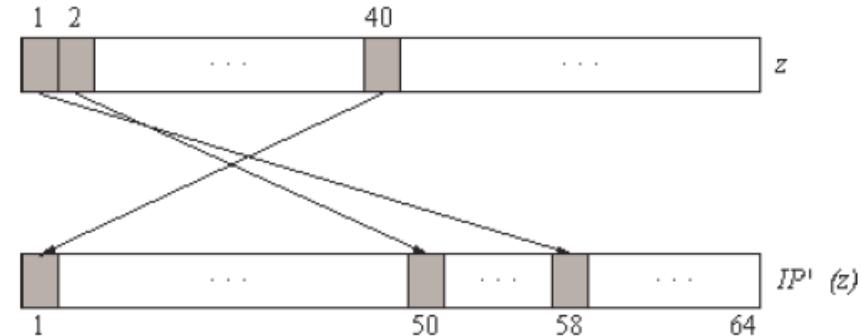
Initiale Permutation

IP							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7



Finale Permutation

$IP^{-1}$							
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

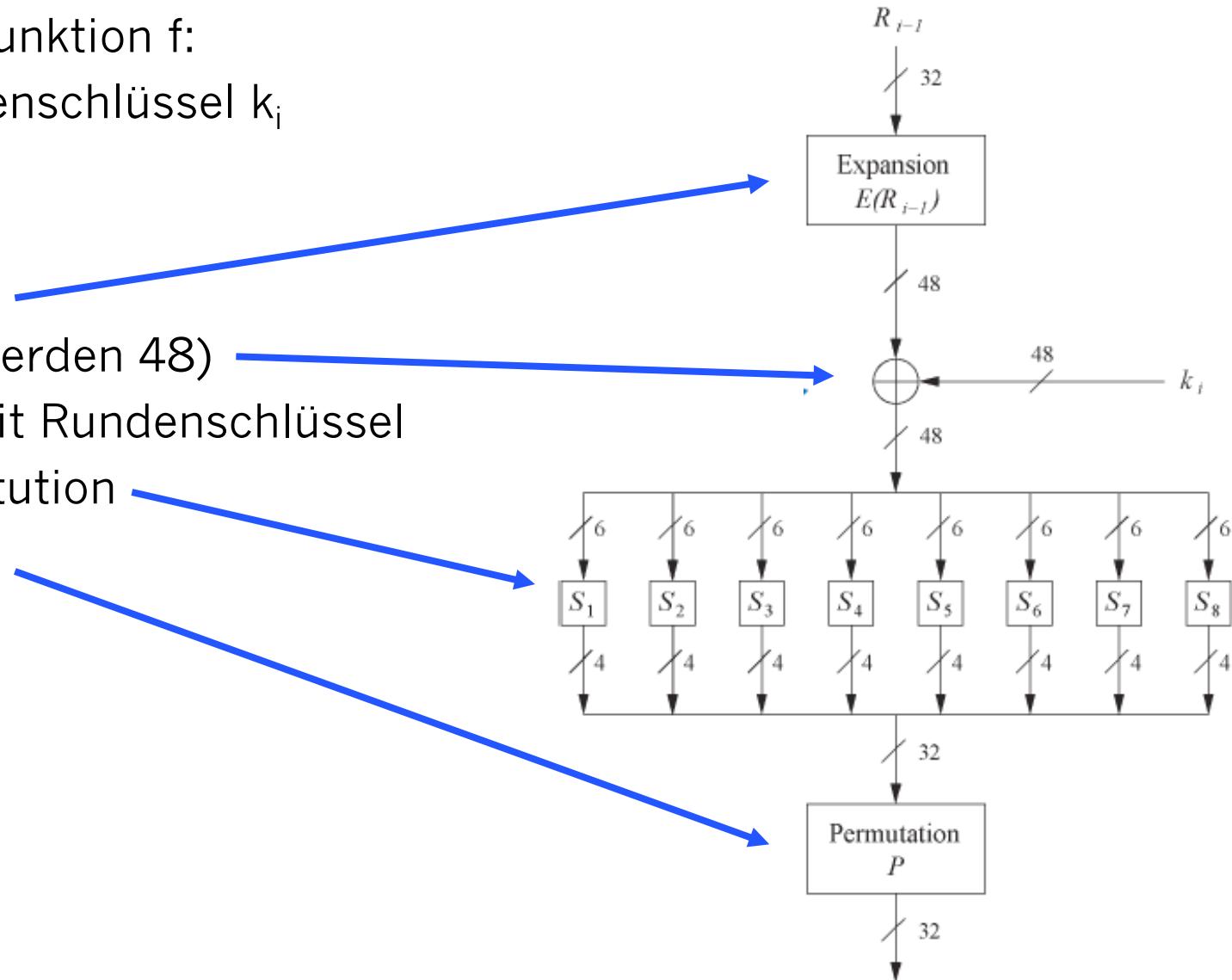


# FUNKTION f

- Eingabe der Funktion f:
- $R_{i-1}$  und Rundenschlüssel  $k_i$

## Vier Schritte

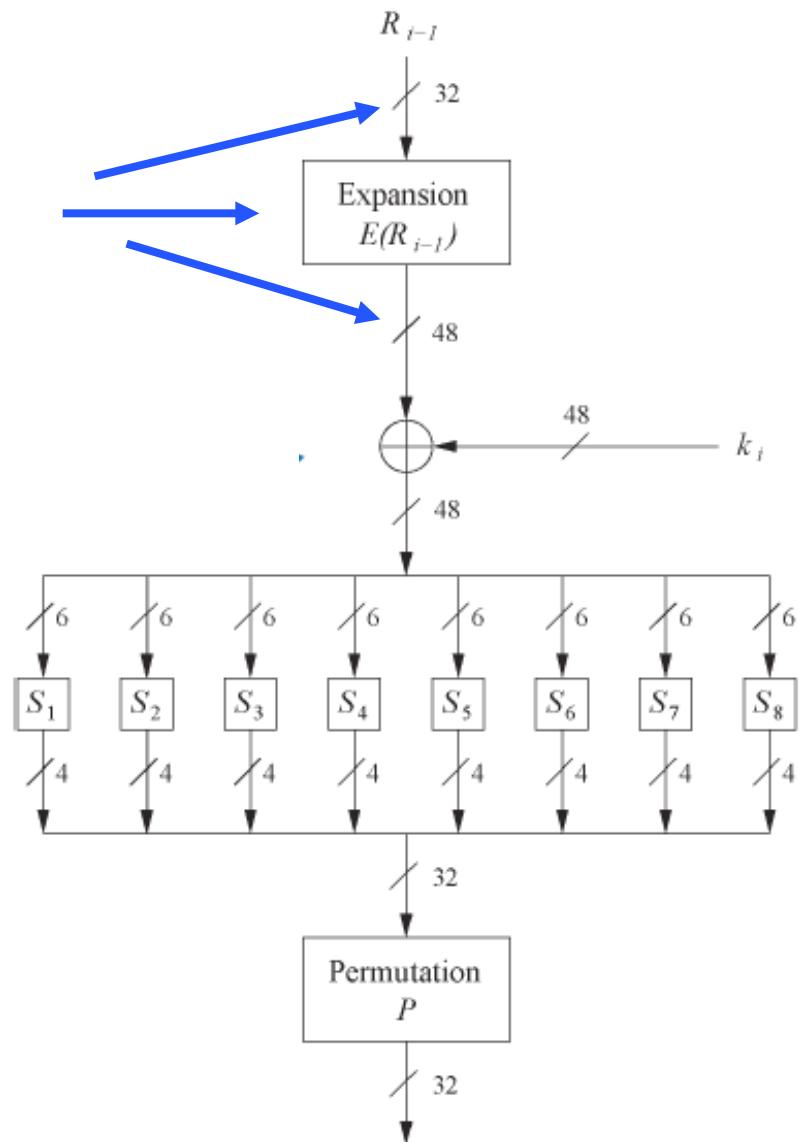
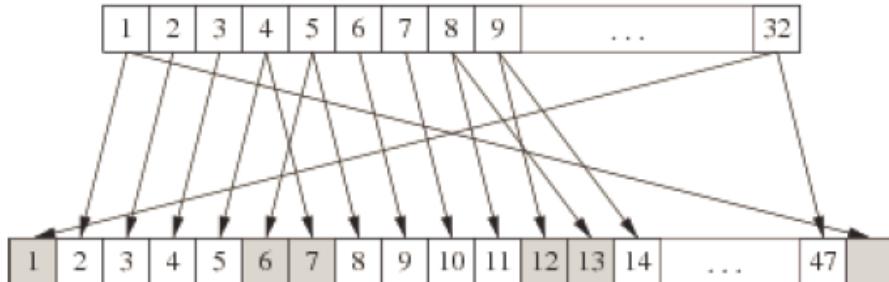
1. Expansion E  
(aus 32 Bit werden 48)
2. XOR (Add) mit Rundenschlüssel
3. S-Box Substitution
4. Permutation



# EXPANSIONSFUNKTION E

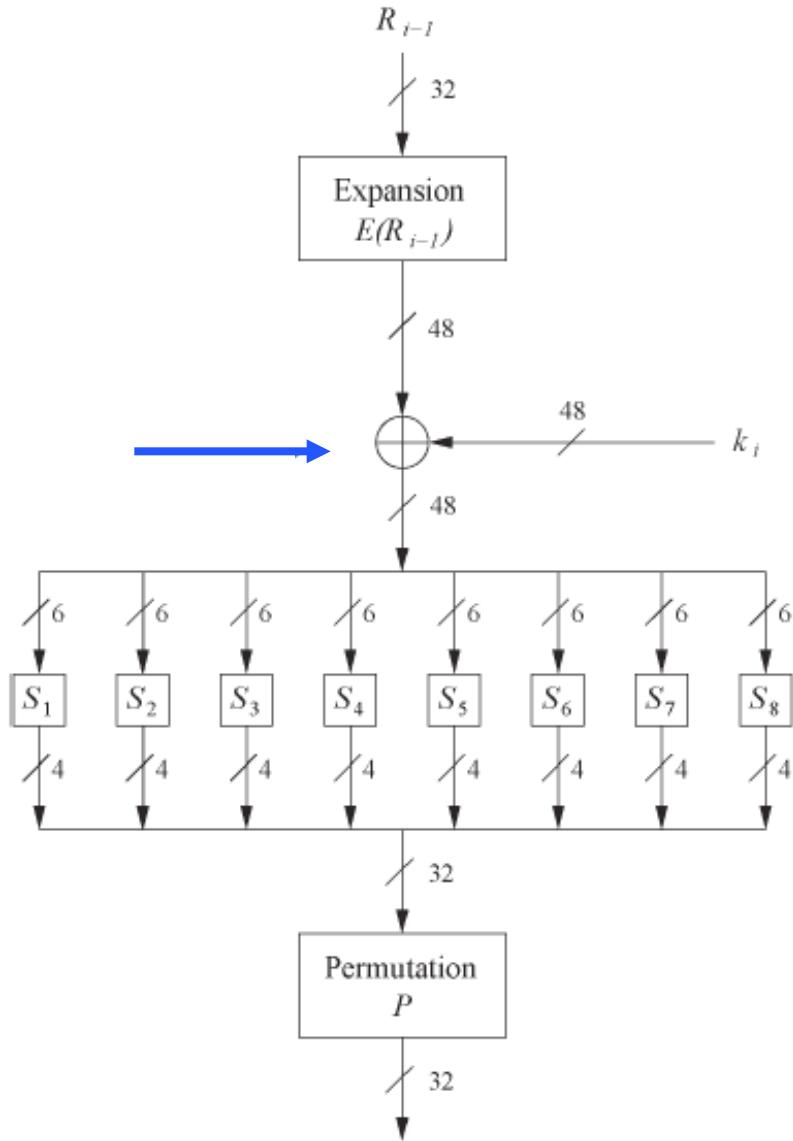
- Hauptzweck:
  - erhöht Diffusion

E					
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1



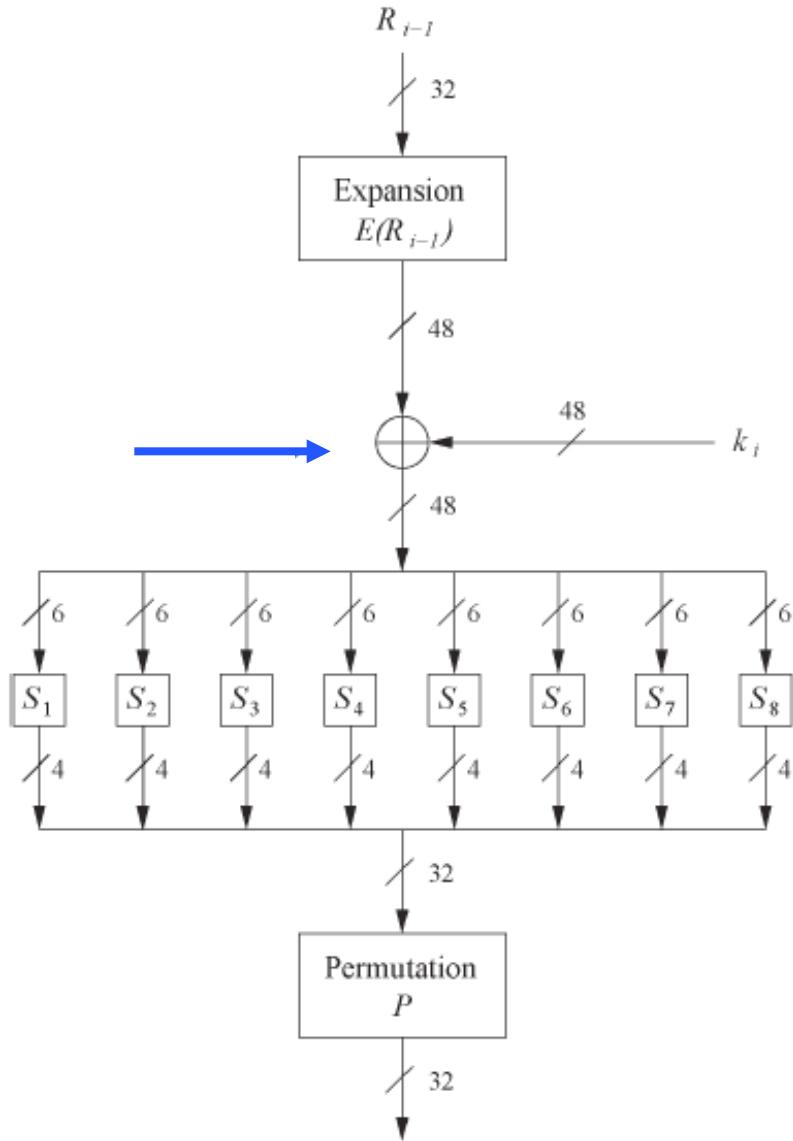
# XOR/ADD ROUNDKEY

- Rundenschlüssel nach spezifischem Schema aus Hauptschlüssel abgeleitet
- Bitweises XOR des Rundenschlüssels mit dem Ergebnis von E



# XOR/ADD ROUNDKEY

- Rundenschlüssel nach spezifischem Schema aus Hauptschlüssel abgeleitet
- Bitweises XOR des Rundenschlüssels mit dem Ergebnis von E



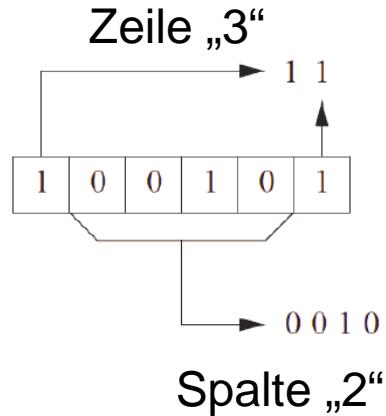
# S-Box SUBSTITUTION

- 8 vorgegebene Substitutionstabellen (S-Box)
- 6 Bits Eingabe, 4 Bits Ausgabe
- Kritische für die Sicherheit von DES
- Spezielle Design-Kriterien für S-Box

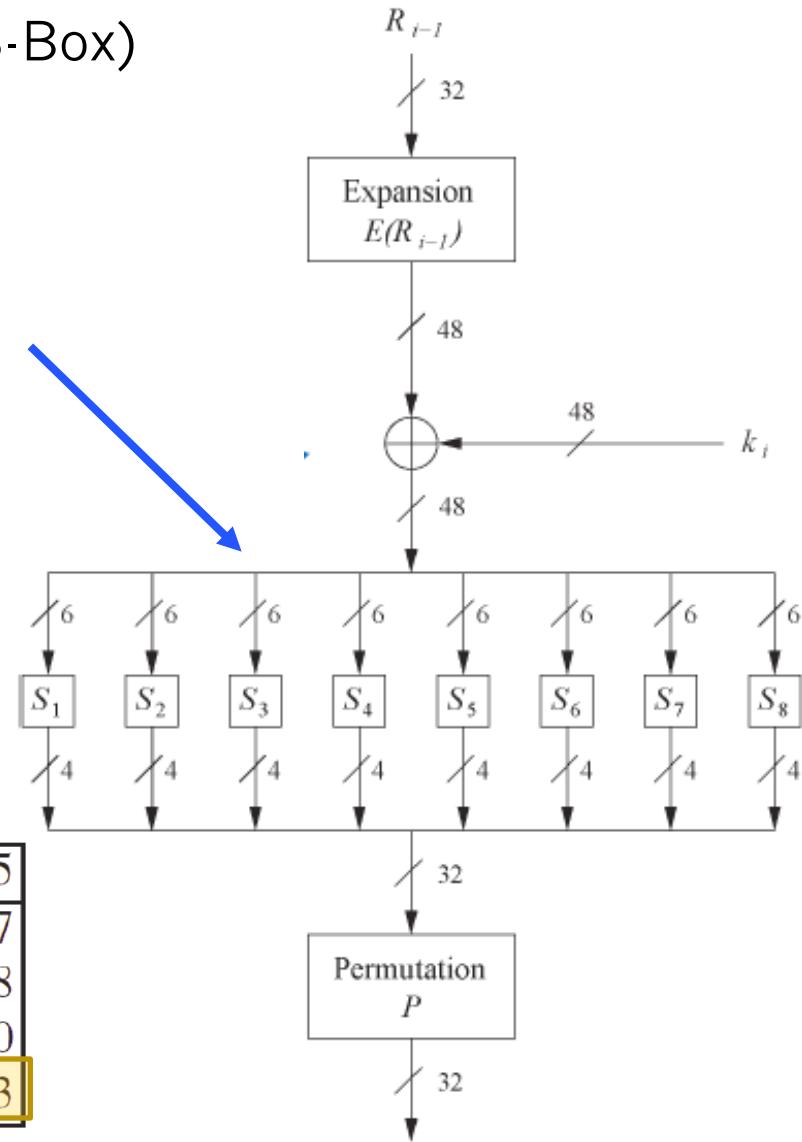
- Beispiel

6 Bit Eingabe

4 Bit Ausgabe



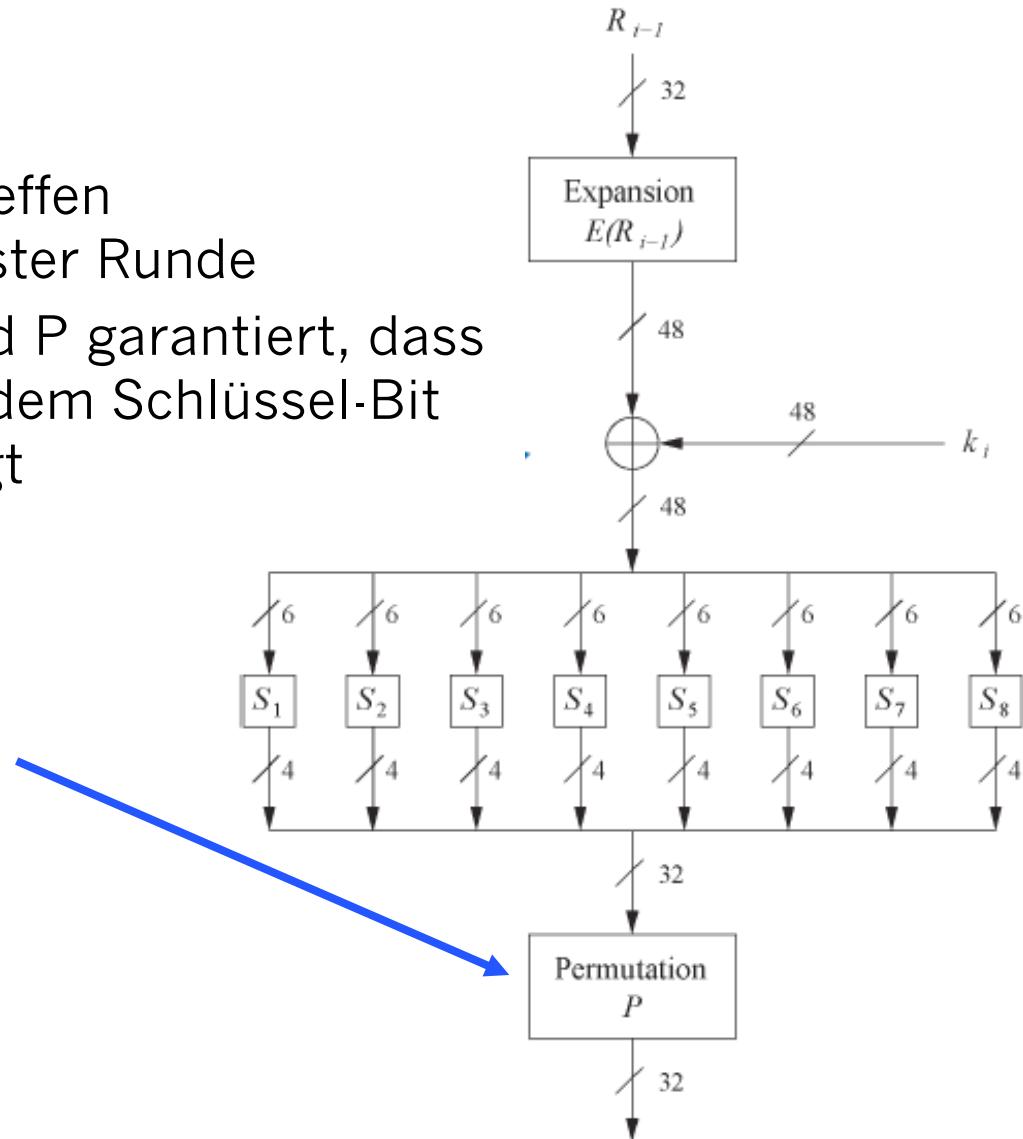
$S_1$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13



# PERMUTATION P

- Bitweise Permutation
- Sorgt für Diffusion
- Ausgabe-Bits einer S-Box betreffen verschiedene S-Boxes in nächster Runde
- Diffusion durch E, S-Boxes und P garantiert, dass nach Runde 5 jedes Bit von jedem Schlüssel-Bit und jedem Klartext-Bit abhängt

P							
16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25



# RUNDENSchlÜSSEL

- Die Eingabeschlüsselgröße von DES ist 64 Bit
  - davon 56 Schlüssel-Bits und 8 Paritätsbits
- Spezifische Vorschrift zur Ableitung von 16 Rundenschlüsseln ki mit je 48 bits aus dem Hauptschlüssel (56 Bit)

55

# SICHERHEIT VON DES

- Bei der Einführung von DES gab es zwei Kritikpunkte
  - Schlüsselraum zu klein (Schlüssel zu kurz mit nur 56 Bit)
  - S-Box Entwurfskriterien wurden/werden geheim gehalten.
    - Gibt es verborgenden analytische Angriffe (Hintertüren), die nur die NSA kennt?
- Analytische Angriffe:
  - DES ist sehr resistent sowohl gegen differentielle als auch lineare Kryptoanalyse – zwei Analyse/Angriffstechniken, die erst Jahre nach DES bekannt/veröffentlicht wurden.
    - ⇒ Das heißt IBM und NSA kannten diese Analyse/Angriffstechniken bereits 15 Jahre lang!
  - Es gibt keinen bekannten analytischen Angriff der DES in realistischen Szenarien bricht.
- Brute Force Angriff:
  - prüfe alle möglichen  $2^{56}$  Schlüssel
  - Relative leicht mit heutiger Rechentechnik durchführbar

56

# BRUTE FORCE ANGRIFFE

- 1998 Deep-Crack-Supercomputer: ca. 250.000 \$
  - Knacken eines DES-Schlüssels in 56 Stunden!
- 2008: COPACOBANA (<http://www.copacobana.org/>):
  - < 10.000 \$, ca 1 Tag zum Knacken eines DES-Schlüssels

57

# FAZIT DES

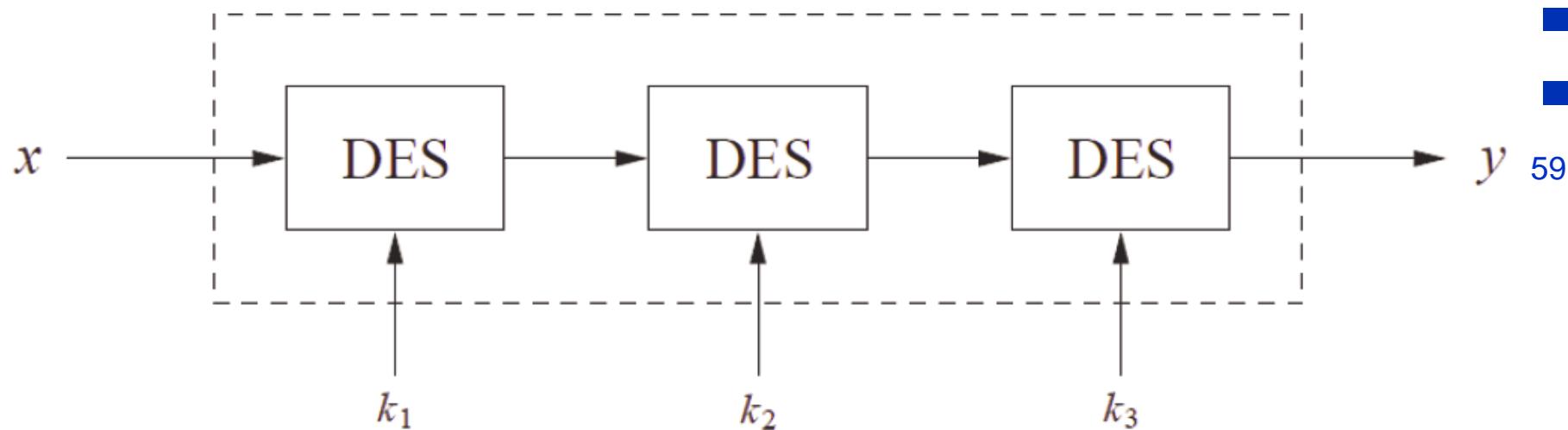
- Stark gegen Kryptoanalysen
- Aber Schlüssellänge zu kurz
- Problem:
  - DES noch immer Bestandteil vieler Anwendungen
  - Austausch von DES gegen stärkeres Verfahren mit längeren Schlüsseln, z.B. AES, ist sehr aufwändig.
- Lösung:
  - Beibehalten des DES, aber Vergrößern des Schlüsselraums!
  - Z.B. 3DES (Tripel-DES) mit mehreren Schlüsseln

58

# TRIPLE DES – 3DES

- Dreifachverschlüsselung mit DES ist weit verbreitet und erweitert die **effektive** Schlüssellänge auf 112 Bits

$$y = DES_{k_3}(DES_{k_2}(DES_{k_1}(x)))$$

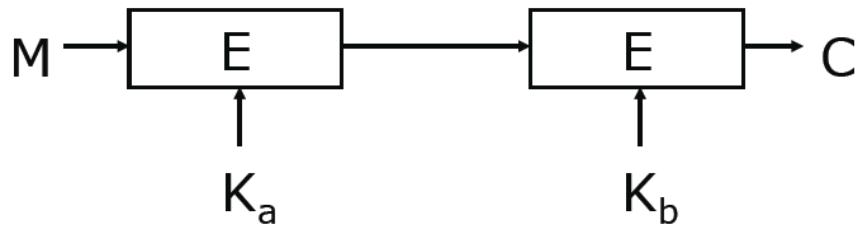


- Alternative Version von 3DES

$$y = DES_{k_3}(DES_{k_2}^{-1}(DES_{k_1}(x))).$$

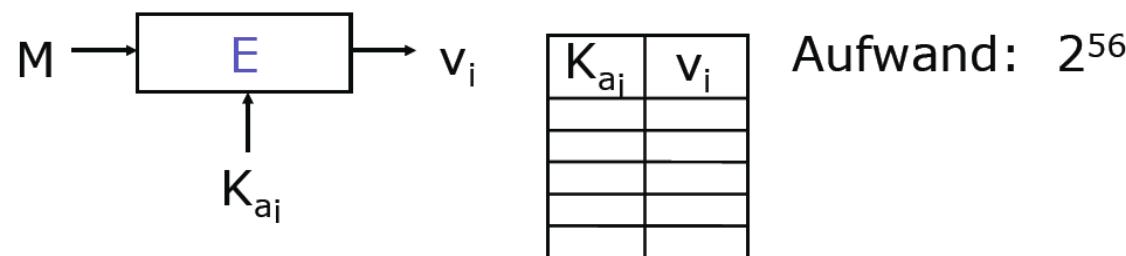
# MÖGLICHER ANGRIFF BEI 2DES – MEET-IN-THE-MIDDLE-ANGRIFF

## ■ Ausgangssituation



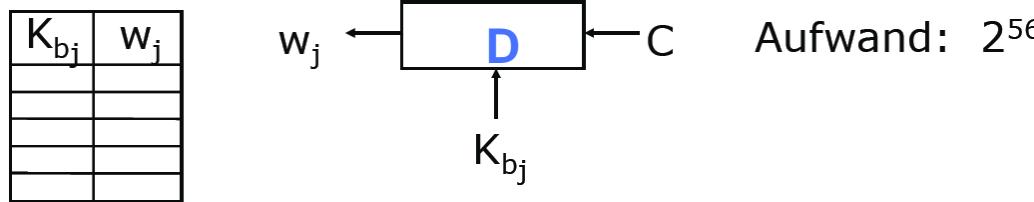
## ■ Known-Plaintext-Angriff

1. Verschlüssele  $M$  für alle mögliche  $K_a$  und speichere die Chiffretexte  $v_i = E(M, K_{aj})$



# MÖGLICHER ANGRIFF BEI 2DES – MEET-IN-THE-MIDDLE-ANGRIFF

- Entschlüssle C für alle mögliche  $K_b$  und speichere die Klartexte  $w_j$  ebenfalls in einer Tabelle  $w_j = D(C, K_{bj})$



- Fall  $v_i == w_j$  für ein bestimmtes Paar i und j, sind  $K_{ai}$  und  $K_{bj}$  die gesuchten Schlüssel, ggf. Probe mit weiteren Klartext/Chiffertextpaaren

- Aufwand
  - $2^{56} + 2^{56} = 2 * 2^{56} = 2^{57}$
  - Sicherheitsgewinn ist nur 1 Bit!
- Nicht DES-spezifisch, sondern gilt für alle Blockchiffren!

# AES – ADVANCED ENCRYPTION STANDARD

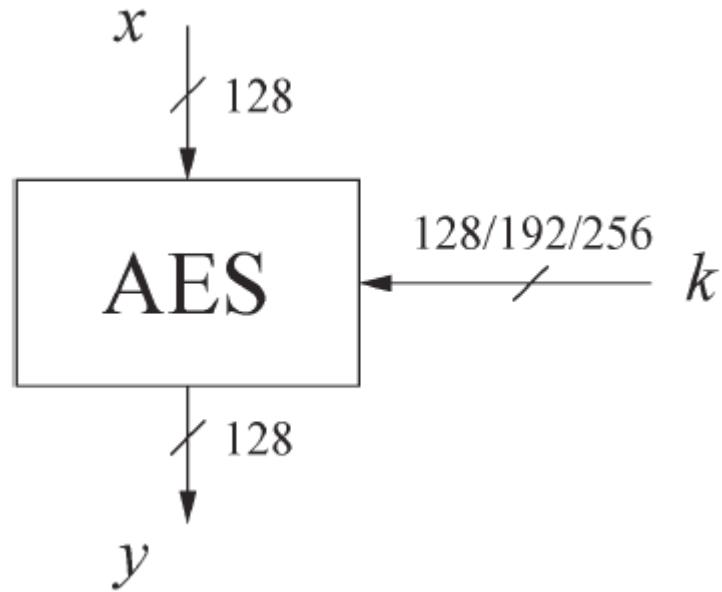
## ■ Basisfakten

- AES ist heute die meist genutzte symmetrische Verschlüsselung
- Der Algorithmus für AES wurde durch das US National Institute of Standards and Technology (NIST) in einem mehrjährigen Auswahlprozess ausgewählt
- Die Anforderungen an alle AES Kandidaten waren
  - Blockchiffre mit **128-bit** Blockgröße
  - unterstützte **Schlüssellängen: 128, 192 und 256 Bit**
  - Sicherheit relativ zu anderen eingereichten Algorithmen
  - Effizienz in Hardware und Software

## ■ Zeitlicher Ablauf der Algorithmenauswahl

- Bedarf an einer neuen Blockchiffre im Januar 1997 vom NIST verkündet
- 15 Kandidaten-Algorithmen im August 1998 akzeptiert
- 5 Finalisten im August 1999:
  - *Mars* – IBM Corporation
  - *RC6* – RSA Laboratories
  - *Rijndael* – J. Daemen & V. Rijmen
  - *Serpent* – Eli Biham et al.
  - *Twofish* – B. Schneier et al.
- Im Oktober 2000 wurde Rijndael als AES ausgewählt
- AES wurde formal als Standard verabschiedet im November 2001

# AES ÜBERBLICK

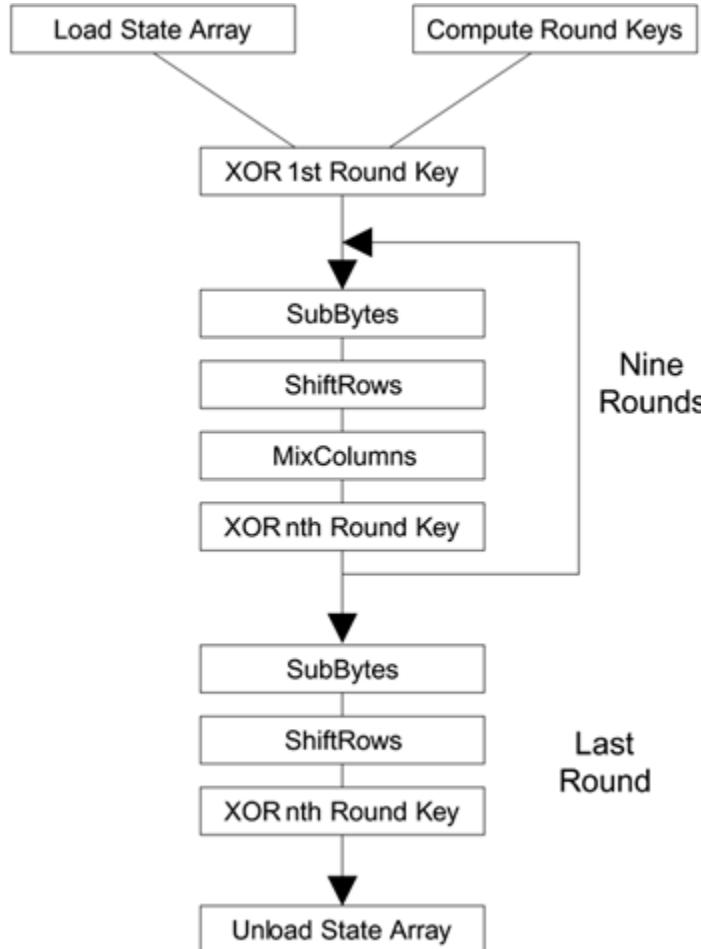


- Anzahl der Runden hängt von der gewählten Schlüssellänge ab

Key length (bits)	Number of rounds
128	10
192	12
256	14

# AES ALGORITHMUS

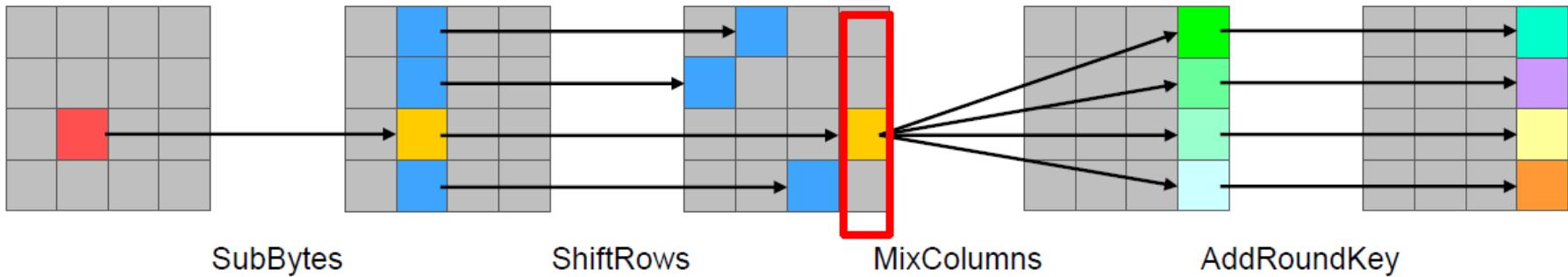
- Eingabeblock als 4x4 Matrix repräsentiert
- AES arbeitet auf Zeilen und Spalten der Matrix
- Matrix wird auch State genannt



# AES ALGORITHMUS

- In jeder Runde werden 4 Transformationsschritte durchlaufen:
  - Round( $\text{State}$ ,  $\text{RoundKey}$ ) {
    - SubBytes( $\text{State}$ ); Substitutionschiffre, byte-weise
    - ShiftRows( $\text{State}$ ); zyklischer Links-Shift
    - MixColumns( $\text{State}$ ); Spaltenweise Multiplikation mit Polynom, jedes Byte der Spalte wird mit jedem anderen der Spalte verknüpft
    - AddRoundKey( $\text{State}$ ,  $\text{RoundKey}$ );
  - }

66



## Wie Sie durch die Anwendung des HijackThis-Addons austasten können:

- o Starten Sie Ihren Browser und das Vollbild anschließen.
- o Benutzer/Passwörter, die Ihnen vorher schon Da weiter zu erhalten.
- o Klicken in dem Disk-Controller unten, um direkt zu einem Drive zu navigieren.
- o Testen Sie, um einen Disk-Controller auszuwählen oder zu verhindern.

# SICHERHEIT VON AES

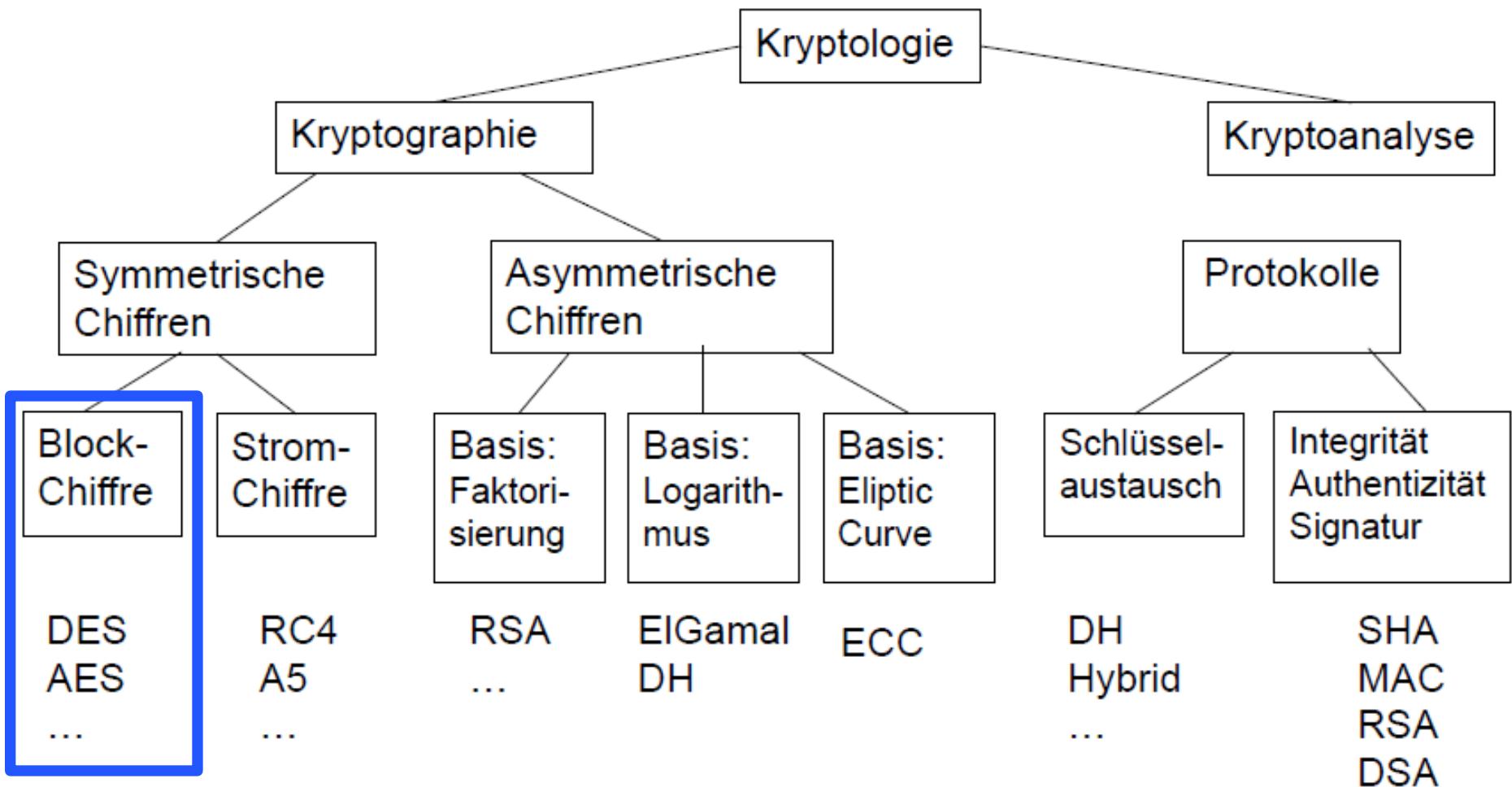
- **Brute-force-Angriffe:** aufgrund der Schlüssellänge von 128, 192 or 256 bits sind Brute-force-Angriffe nicht möglich
- **Analytische Angriffe:** es ist keine analytischer Angriff bekannt, der besser als Brute Force ist
- **Seitenkanal-Angriffe**
  - Es wurden Seitenkanal-Angriffe veröffentlicht
  - Seitenkanalangriffe sind jedoch keine Angriffe gegen den zugrundeliegenden Algorithmus sondern eine Implementierung desselben

68

# FRAGEN?

69

# DIAGRAMM



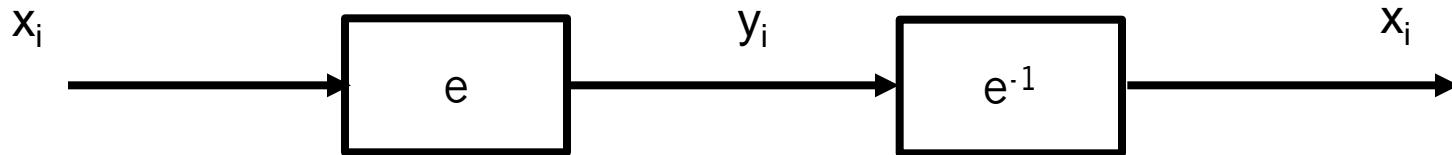
# VERSCHLÜSSELUNG MIT BLOCKCHIFFREN

- Zur Erinnerung Blockgröße von
  - DES: 64 Bit
  - AES: 128, 196 oder 256 Bit
- Es gibt verschiedene Möglichkeiten große Klartexte (E-Mail, Datei) mit einer Blockchiffre zu verschlüsseln
- ⇒ Betriebsmodi (für Blockchiffren) aka Blockmodi
  - Electronic Code Book mode (ECB)
  - Cipher Block Chaining mode (CBC)
  - Output Feedback mode (OFB)
  - Cipher Feedback mode (CFB)
  - Counter mode (CTR)
- Alle Modi haben das Ziel, zusätzlich zur Vertraulichkeit auch Authentizität und Integrität zu sichern
  - Kommt die Nachricht tatsächlich vom ursprünglichen Sender? (Authentizität)
  - Wurde der Chiffretext während der Übertragung verändert? (Integrität)

# ELECTRONIC CODE Book MODE (ECB)

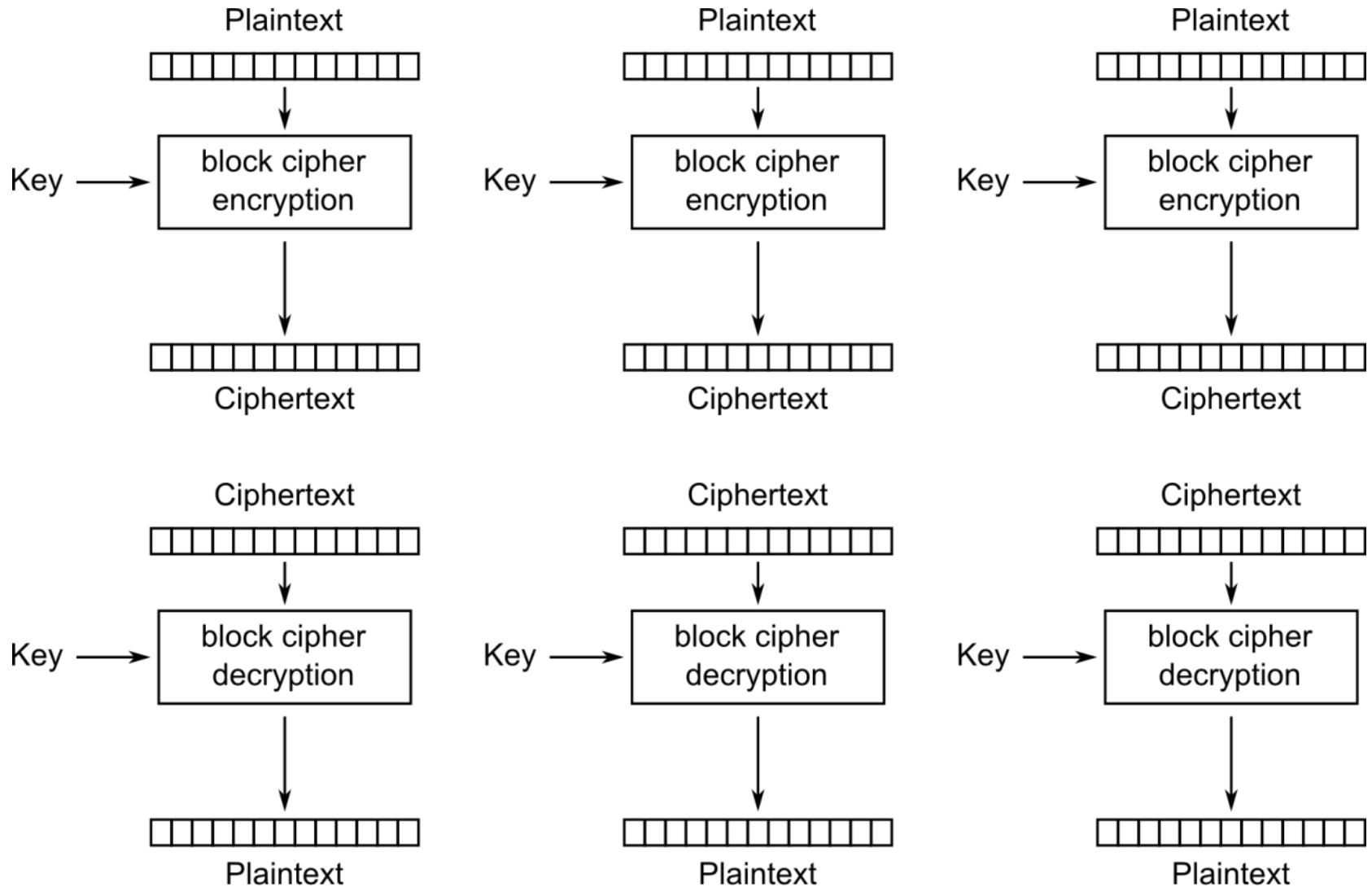
- $e_k(x_i)$  beschreibt die Verschlüsselung eines  $b$ -Bit-Klartextblockes  $x_i$  mit Schlüssel  $k$
- $e_k^{-1}(y_i)$  beschreibt die Entschlüsselung des  $b$ -Bit-Chiffertextblockes  $y_i$  mit Schlüssel  $k$
- Nachrichten größer als  $b$  Bit werden in  $b$ -Bit-Blöcke eingeteilt
- Jeder Block wird separat verschlüsselt

72



- Verschlüsselung:  $y_i = e_k(x_i)$ ,  $i \geq 1$
- Entschlüsselung:  $x_i = e_k^{-1}(y_i) = e_k^{-1}(e_k(x_i))$ ,  $i \geq 1$

# ECB MODE – VER- UND ENTSchlÜSSELUNG



# ECB: VOR- UND NACHTEILE

## ■ Vorteile

- Keine Blocksynchronisation zwischen Sender und Empfänger erforderlich
- Bit-Fehler während der Übertragung betreffen nur den jeweiligen Block aber keine folgenden Blöcke
- Betrieb der Blockchiffre kann parallelisiert werden
  - Vorteil für Hochleistungsimplementierung

## ■ Nachteile

- ECB verschlüsselt hoch deterministisch
  - Identische Klartexte resultieren in identischen Chiffretexten
  - Ein Angreifer erkennt wenn dieselbe Nachricht zweimal gesendet wurde
  - Klartextblöcke werden unabhängig von Vorgängerblöcken verschlüsselt
    - Ein Angreifer kann die Chiffretextblöcke umordnen und es resultieren valide Klartextblöcke

# SUBSTITUTIONS-ATTACKE AUF ECB

- Sobald ein Klartext-Chiffretext-Paar ( $x_i, y_i$ ) bekannt ist, kann Sequenz von Chiffretext-Blöcken leicht manipuliert werden
- Beispiel Elektronische Überweisung

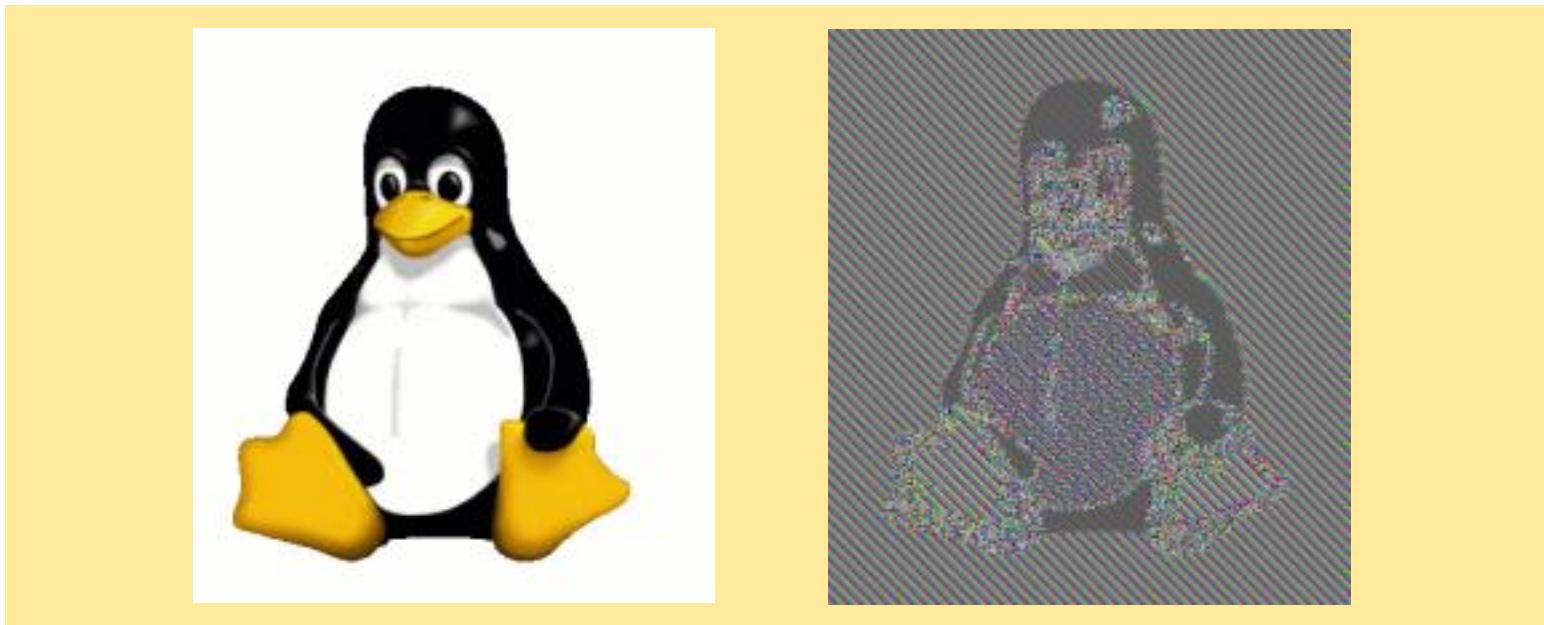
Blocknr.:	1	2	3	4	5
	Sender-Bank A	Sender-Konto-Nr.	Empfänger-Bank B	Empfänger-Konto-Nr.	Betrag €

75

- Annahme: Verschlüsselungsschlüssel zwischen zwei Banken ändern sich nicht allzu häufig
- Angreifer überweist 1€ von seinem Konto bei Bank A auf sein Konto bei Bank B
  - Er kann prüfen, welchen Chiffretextblöcke sich wiederholen und er speichert Blöcke 1, 3 und 4 dieser Überweisungen
- Angreifer ersetzt Block 4 anderer Überweisungen mit zuvor gespeichertem Block 4
  - Alle Überweisungen von Konten bei Bank A an Konten bei Bank B werden umgeleitet auf das Angreiferkonto bei Bank B

# BEISPIEL ZUR VERSCHLÜSSELUNG VON BITMAPS IM ECB MODUS

- Eine Grafik mit nur wenigen schwarzen Strichen auf weißem Grund wobei Bitwert 1 schwarz und Bitwert 0 weiß kodiert enthält viele Blöcke die nur aus 0 bestehen
  - All diese Blöcke haben identischen Chiffretextblock
  - Dadurch Grafik rekonstruierbar ohne Schlüssel zu kennen

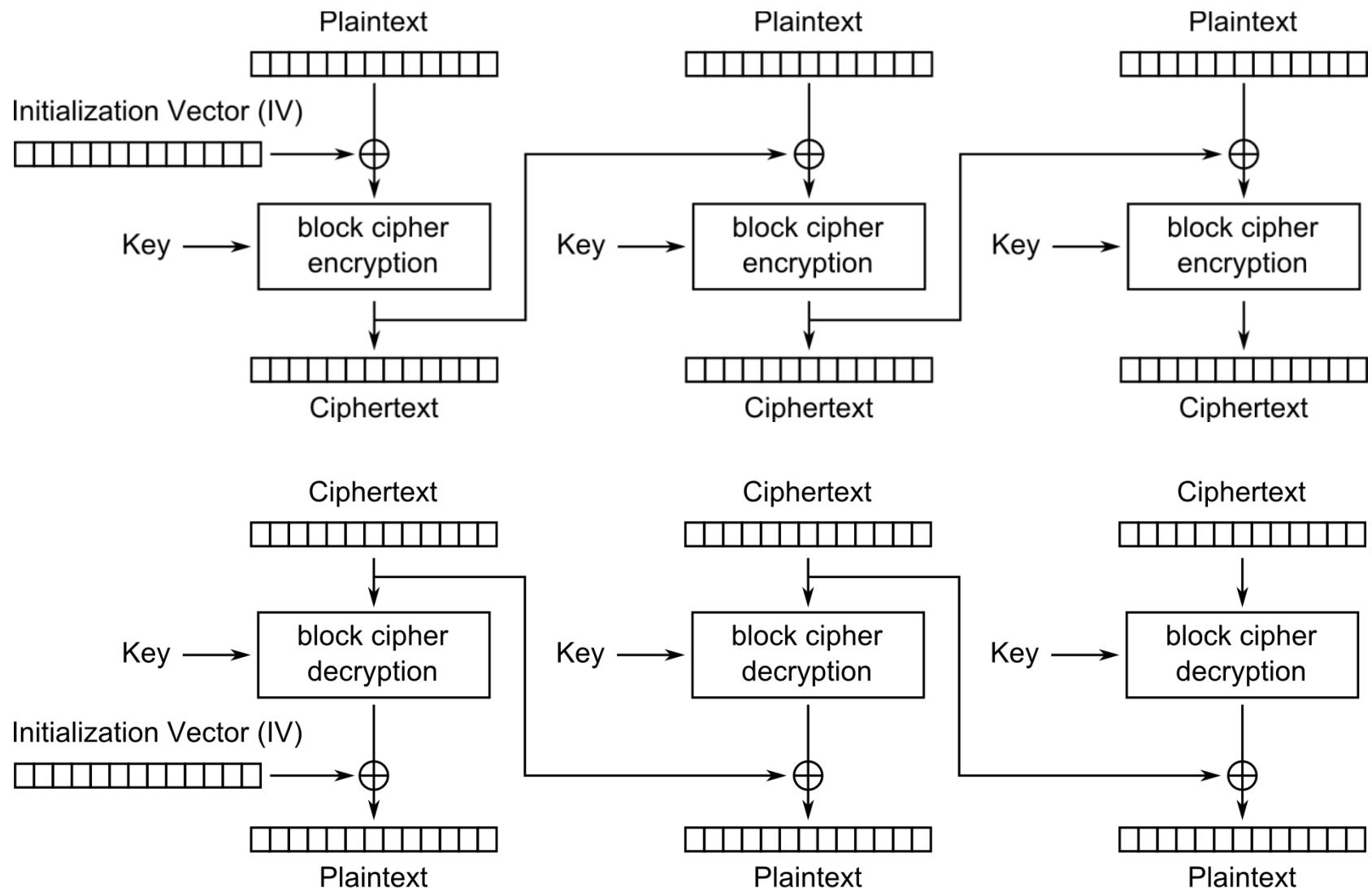


- Einheitliche und großflächige Bereich die sich über mehrere Blöcke erstrecken sind im Chiffretext noch erkennbar

# CIPHER BLOCK CHAINING MODE (CBC)

- Zwei Hauptideen hinter dem CBC Mode
  - Die Verschlüsselung aller Blöcke wird „zusammen gekettet“
    - Chiffertext  $y_i$  hängt nicht nur vom Klartextblock  $x_i$  sondern allen vorangegangen Blöcken ab.
  - Die Verschlüsselung wird randomisiert mittels eines Initialisierungsvektors (IV)
- Verschlüsselung (erster Block):  $y_1 = e_k(x_1 \oplus IV)$
- Verschlüsselung (allgemeiner Block):  $y_i = e_k(x_i \oplus y_{i-1}), i \geq 2$
- Entschlüsselung (erster Block):  $x_1 = e_k^{-1}(y_1) \oplus IV$
- Entschlüsselung (allgemeiner Block):  $x_i = e_k^{-1}(y_i) \oplus y_{i-1}, i \geq 2$

# CBC MODE – VER- UND ENTSchlÜSSELUNG



# CBC MODE

- Für den ersten Klartextblock  $x_1$  gibt es keinen vorherigen Chiffretextblock
- ein Initialisierungsvektor IV wird zum ersten Klartextblock hinzugefügt, um jede CBC-Verschlüsselung nicht-deterministisch zu machen
  - Nicht-deterministisch bzw. probabilistisch heißt: zwei Verschlüsselungen des gleichen Klartextes sehen komplett verschieden aus
- Der erste Chiffretextblock  $y_1$  hängt vom Klartextblock  $x_1$  und dem IV ab
- Der zweite Chiffretextblock  $y_2$  hängt von IV,  $x_1$  und  $x_2$  ab
- Der dritte Chiffretextblock  $y_3$  hängt von IV und  $x_1$ ,  $x_2$  und  $x_3$  an, und so weiter ...

# SUBSTITUTIONS-ATTACKE AUF CBC

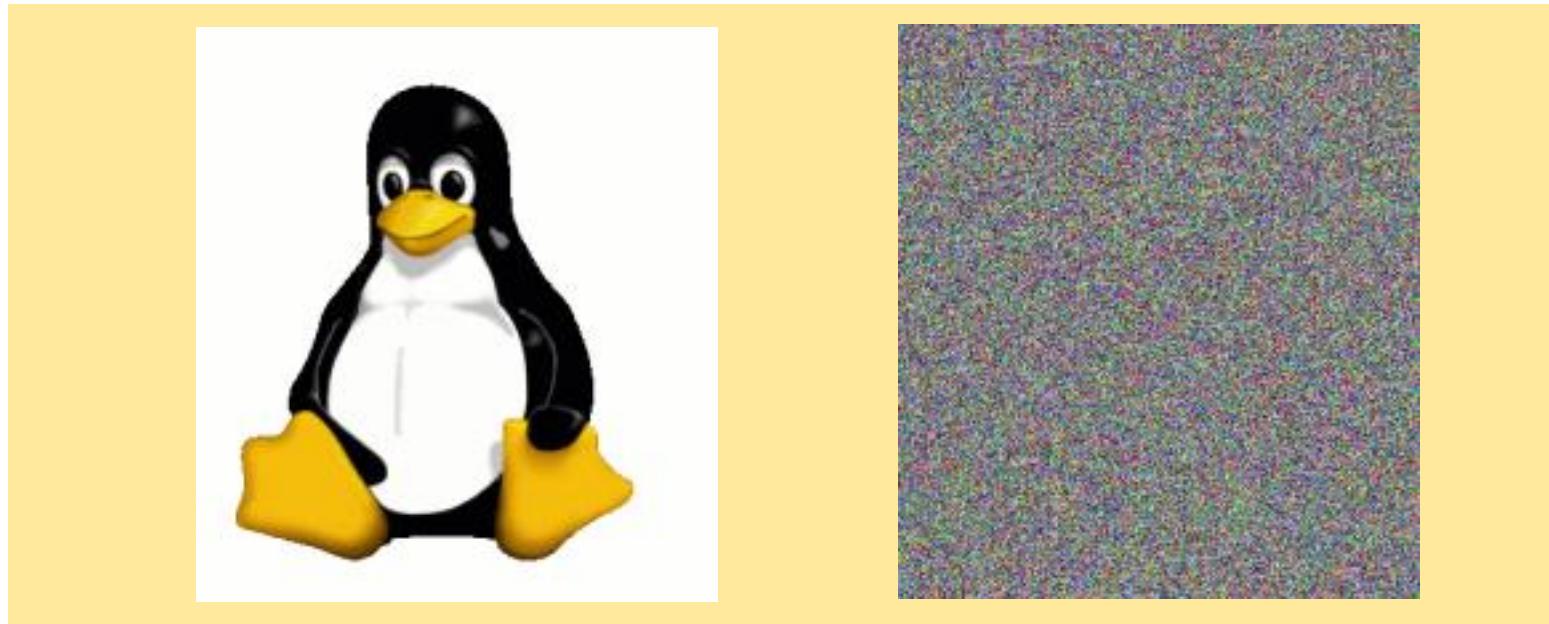
- Beispiel Elektronische Überweisung (wie gehabt)

Blocknr.:	1	2	3	4	5
	Sender-Bank A	Sender-Konto-Nr.	Empfänger-Bank B	Empfänger-Konto-Nr.	Betrag €

- Wenn IV für jede Überweisung neu gewählt wird, dann wird die Attacke nicht funktionieren.
- Wenn ein IV für mehrere Überweisungen genutzt wird, dann kann der Angreifer Überweisungen von seinem Konto bei Bank A zu Bank B erkennen
- Wenn IV bei jeder Verschlüsselung neu gewählt wird, dann wird der CBC Modus zu einem probabilistischen Verschlüsselungsschema
- Es ist nicht notwendig den IV geheim zu halten!
- Typischerweise sollte der IV eine nicht-geheime Nonce sein.
- Nonce: Nicht-geheimer Wert der nur einmal verwendet wird.

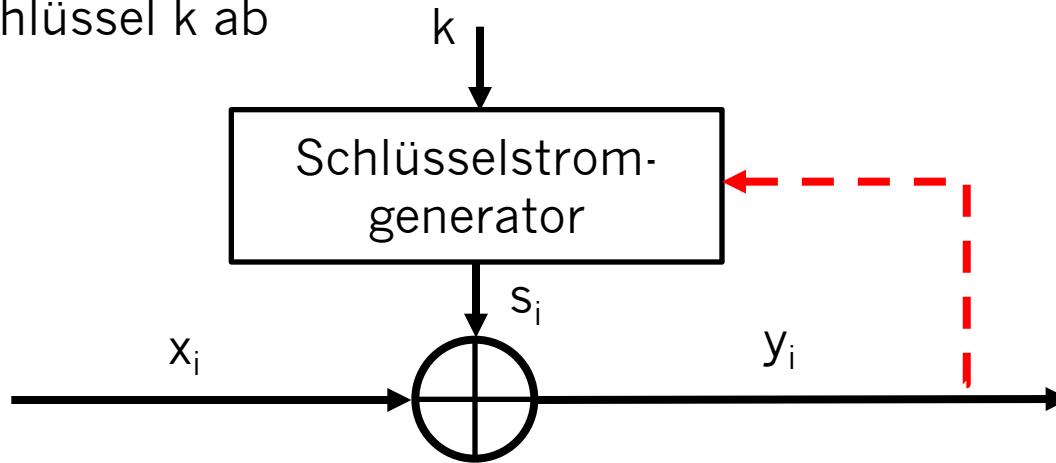
80

# VERSCHLÜSSELUNG VON BITMAPS IM CBC MODUS



81

- Die Sicherheit von Stromchiffren hängt ausschließlich vom Schlüsselstrom  $s_i$  ab
  - sollte zufällig sein, d.h.  $\Pr(s_i=0) = \Pr(s_i=1) = 0,5$
  - muss reproduzierbar sein bei Sender und Empfänger
  - hängt von Schlüssel  $k$  ab



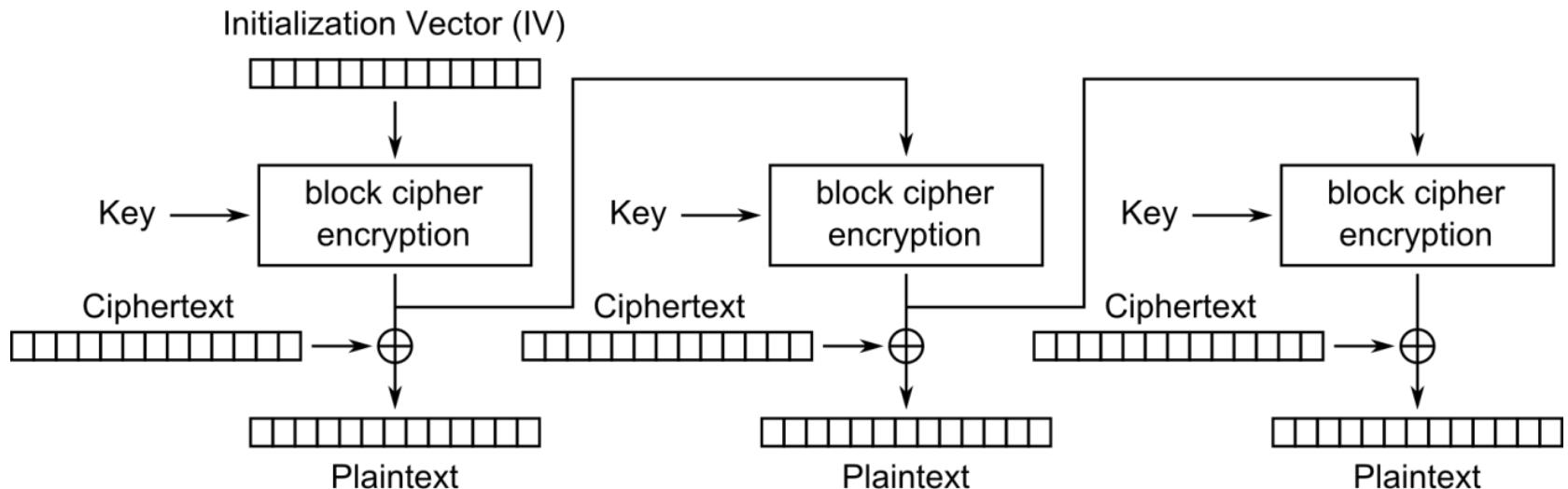
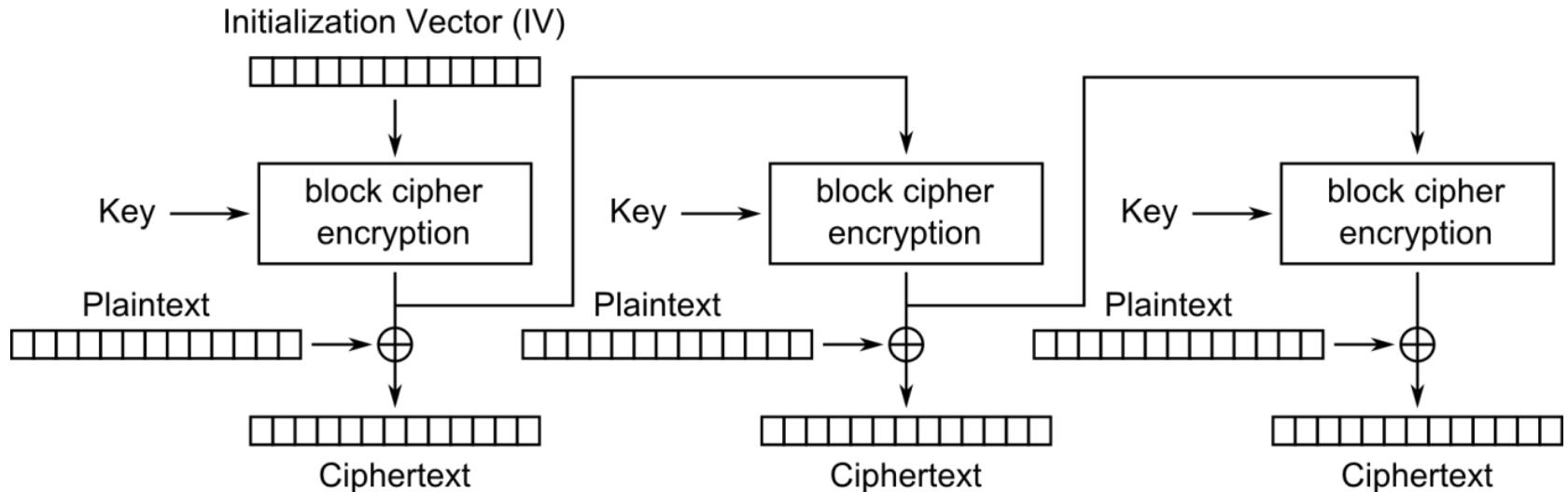
- Synchrone Stromchiffre
  - Schlüsselstrom hängt nur vom Schlüssel ab (und ggf. einem Initialisierungsvektor IV))
- Asynchrone Stromchiffre
  - Schlüsselstrom hängt auch vom Chiffretext ab (---)

# OUTPUT FEEDBACK Modus (OFB)

- Wird benutzt um aus einer Blockchiffre eine synchrone Stromchiffre zu konstruieren
  - Schlüsselstrom wird nicht bitweise sondern blockweise generiert
  - Ausgabe liefert Schlüsselstrombits  $s_i$  mit denen Klartextbits durch XOR verschlüsselt werden können
- 
- Verschlüsselung (1. Block):  $s_1 = e_k(IV)$  und  $y_1 = s_1 \oplus x_1$
  - Verschlüsselung (allg. Block):  $s_i = e_k(s_{i-1})$  und  $y_i = s_i \oplus x_i$ ,  $i \geq 2$
  - Entschlüsselung (1. Block):  $s_1 = e_k(IV)$  und  $x_1 = s_1 \oplus y_1$
  - Entschlüsselung (allg. Block) :  $s_i = e_k(s_{i-1})$  und  $x_i = s_i \oplus y_i$ ,  $i \geq 2$
- 
- es wird nur  $e$  und kein  $e^{-1}$  verwendet/benötigt
  - **Nicht-selbstsynchronisierend:** entschlüsselnder Empfänger benötigt gleichen Zustand wie verschlüsselnder Sender.
    - Genaues zeitliches Zusammenspiel muss sichergestellt werden!

83

# OFB – VER- UND ENTSchlÜSSELUNG



# CIPHER FEEDBACK MODE (CFB)

- Nutzt Blockchiffre als Baustein für eine asynchrone Stromchiffre (ähnlich OFB mode), exakterer Name: “Ciphertext Feedback Mode”
- Schlüsselstrom  $S_i$  wird blockweise generiert und ist abhängig vom Chiffretext
- Resultierend aus der Verwendung eines IV ist die CFB Verschlüsselung auch nicht-deterministisch

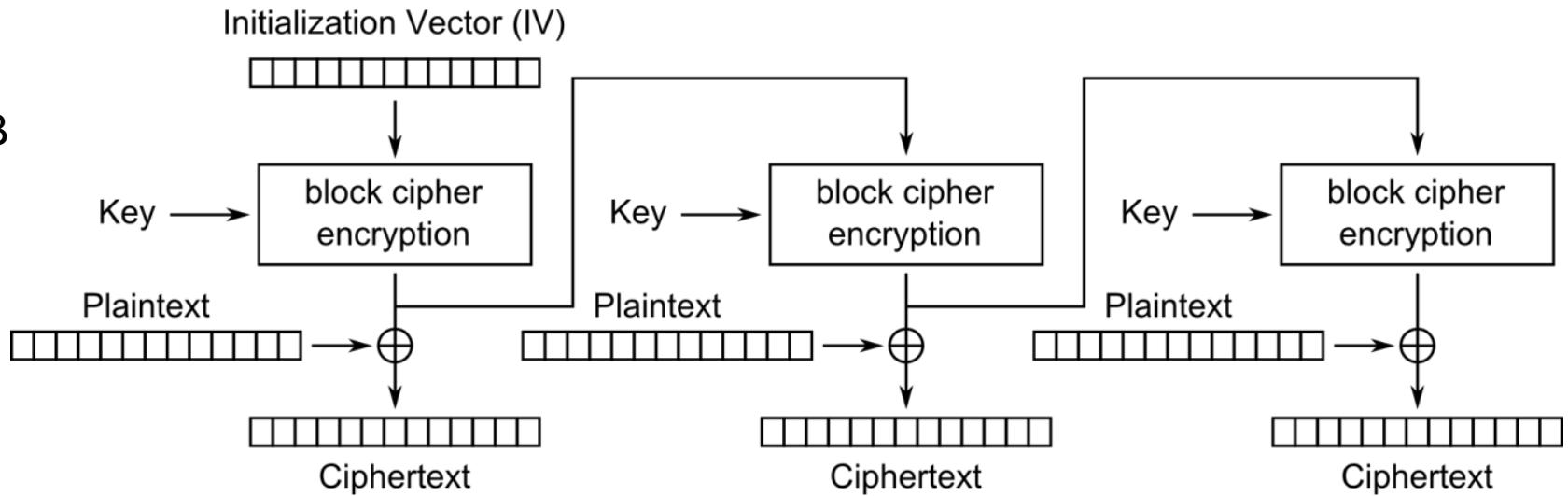
85

- Verschlüsselung (1. Block):  $y_1 = e_k(\text{IV}) \oplus x_1$
- Verschlüsselung (allg. Block):  $y_i = e_k(y_{i-1}) \oplus x_i, i \geq 2$
- Entschlüsselung (1. Block):  $x_1 = e_k(\text{IV}) \oplus y_1$
- Entschlüsselung (allg. Block):  $x_i = e_k(y_{i-1}) \oplus y_i, i \geq 2$
- **Selbstsynchronisierend** (anhand der Chiffretextböcke)!

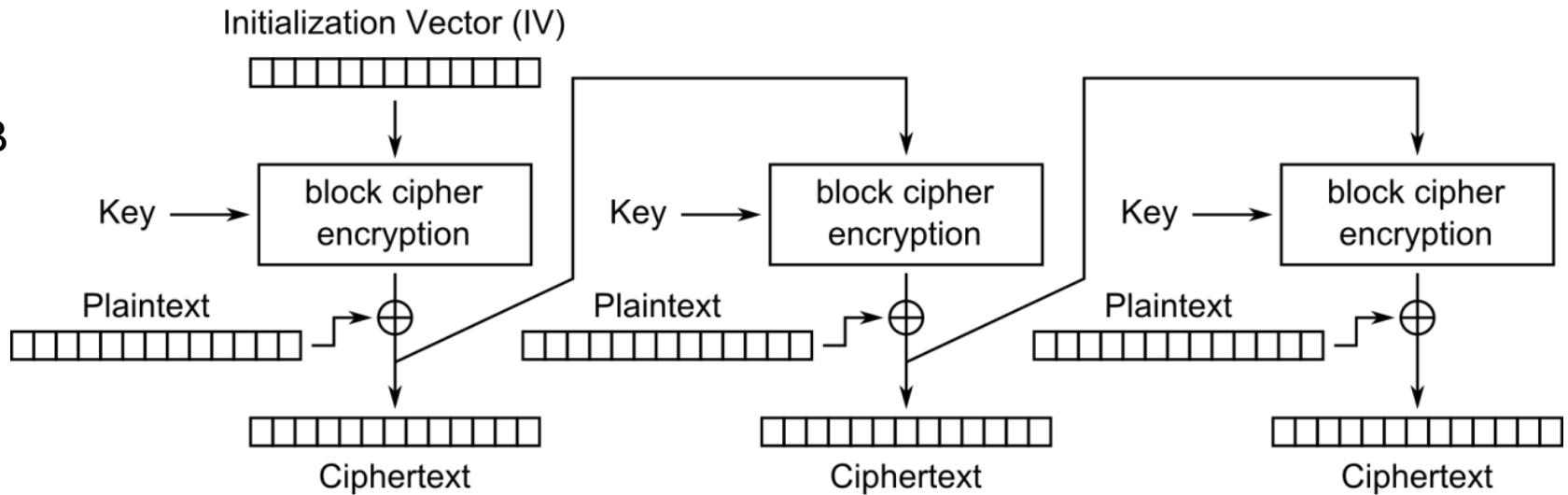
# OFB vs. CFB VERSCHLÜSSELUNG

86

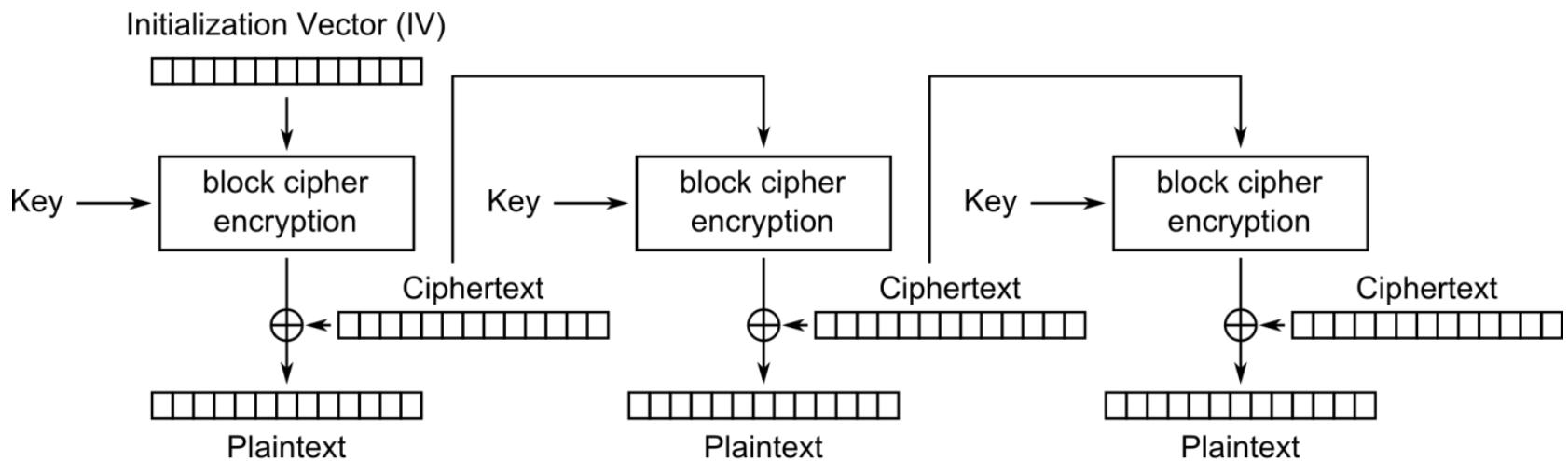
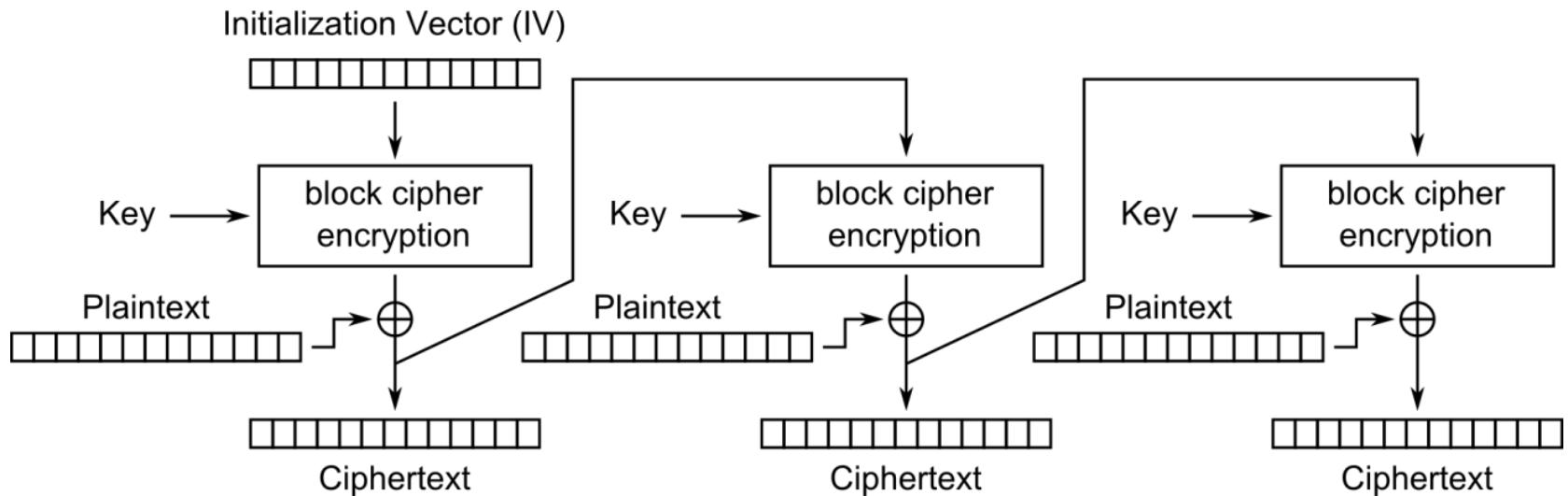
OFB



CFB



# CFB – VER- UND ENTSchlÜSSELUNG



# COUNTER MODE (CTR)

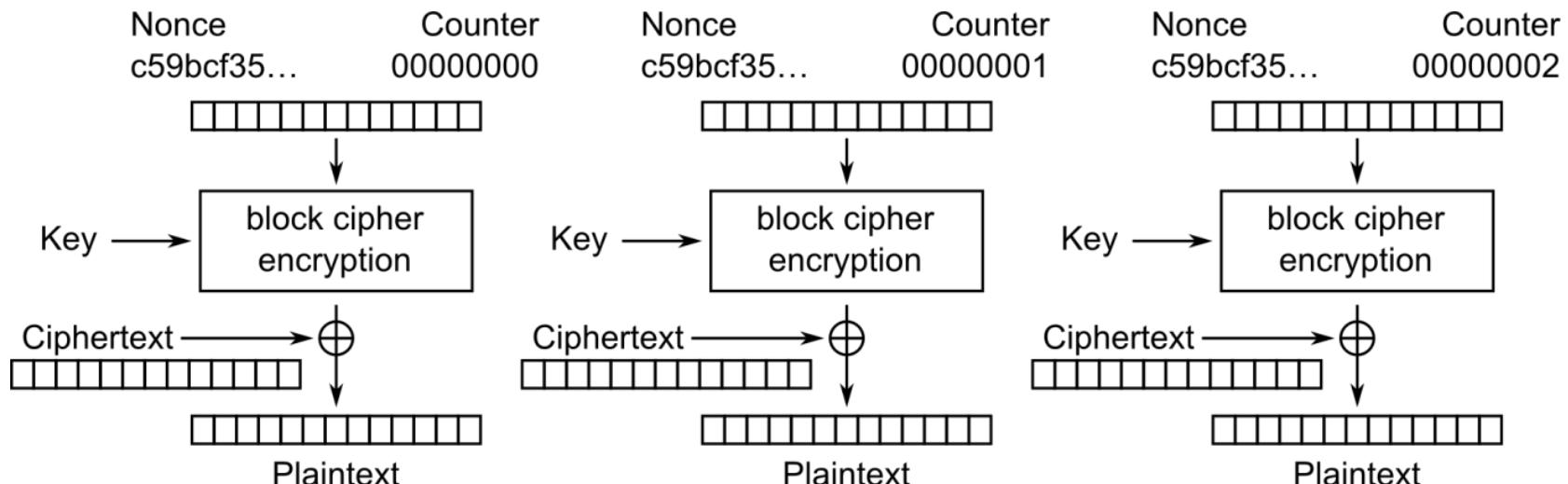
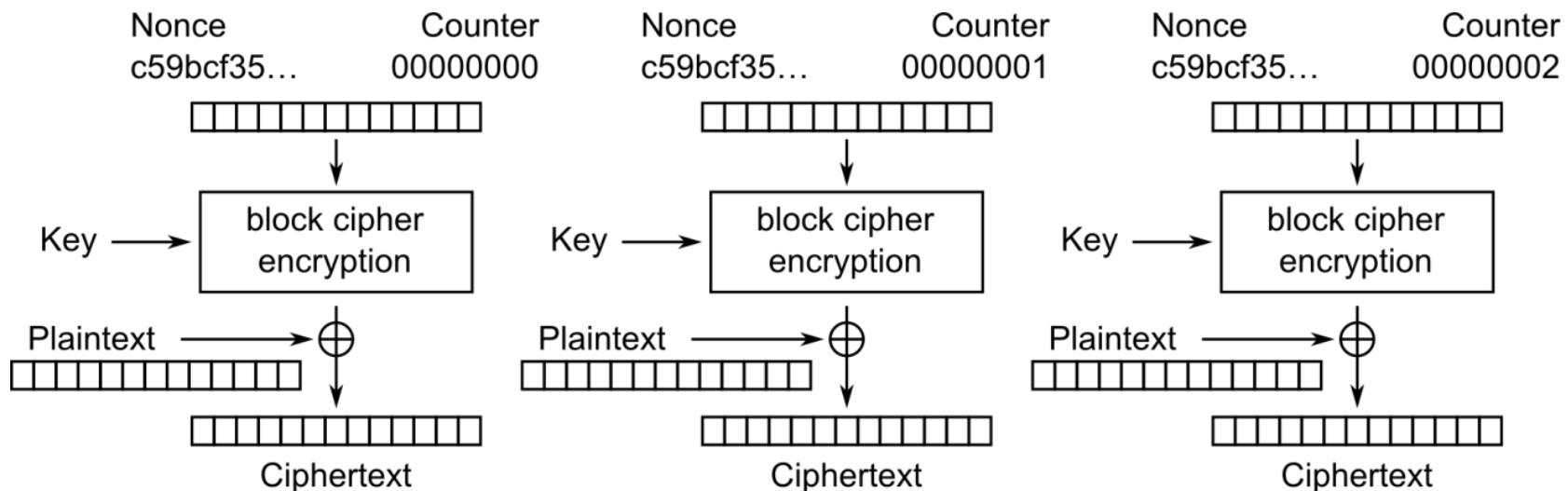
- Verwendet Blockchiffre als Stromchiffre (wie OFB und CFB)
- Der Schlüsselstrom wird blockweise berechnet
- Eingabe für die Blockchiffre ist ein Zähler der mit jedem weiteren Block hochgezählt wird, also verschieden ist jedes mal wenn ein Schlüsselstromblock berechnet wird.
- Initialisierungsvektor (Nonce) für jede Verschlüsselung neu gewählt

Konkatenation o.ä.

- **Verschlüsselung:**  $y_i = e_k(IV \parallel CTR_i) \oplus x_i, i \geq 1$
- **Entschlüsselung :**  $x_i = e_k(IV \parallel CTR_i) \oplus y_i, i \geq 1$
- Anders als beim CFB und OFB Mode kann der CTR Mode parallelisiert werden, da Verschlüsselung vom 2. Block beginnen kann bevor Verschlüsselung vom 1. beendet ist.
  - ⇒ Geeignet für Hochgeschwindigkeits-Implementierung, z.B. in Netzwerk-Routern

88

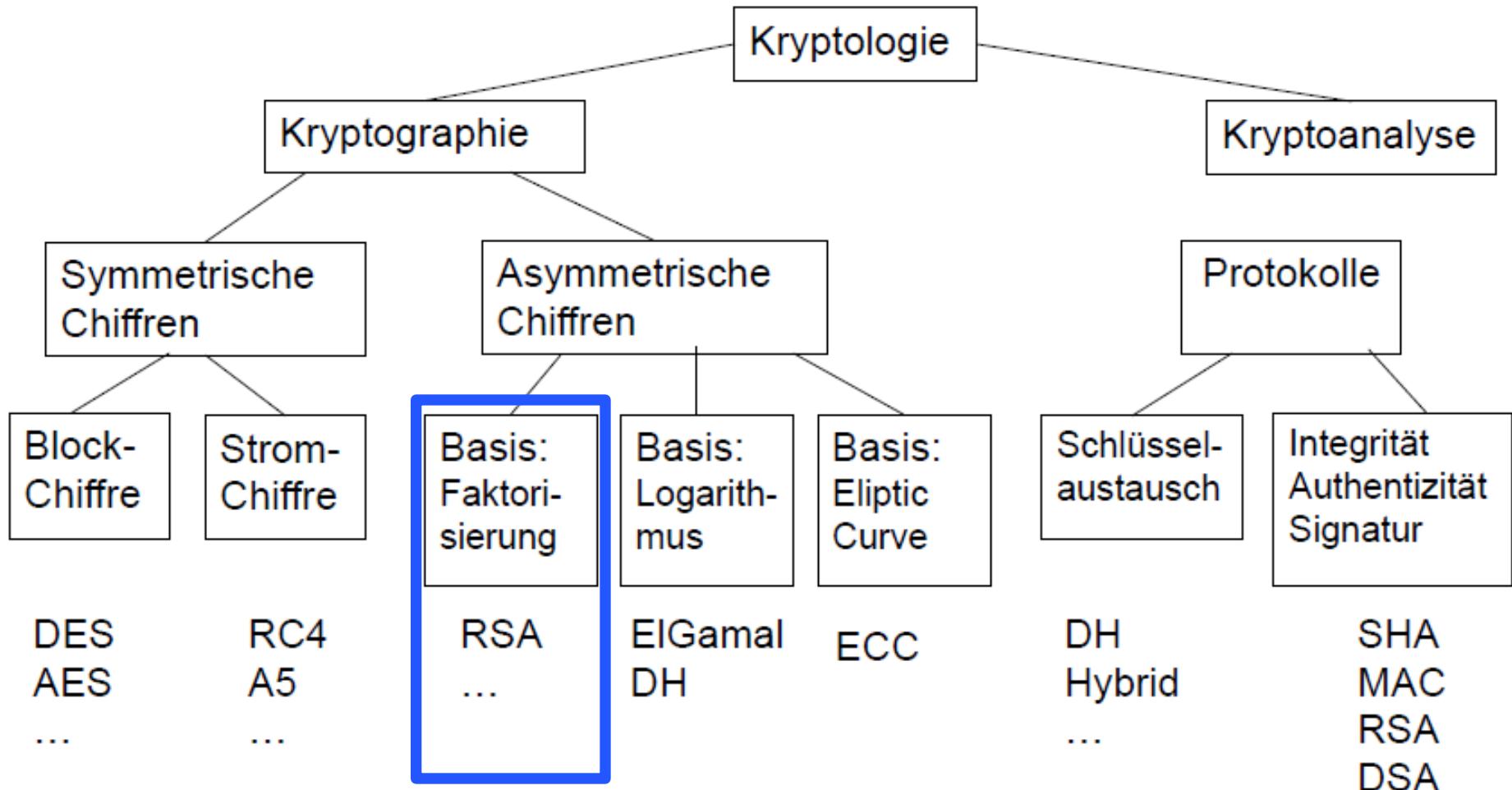
# CTR MODE - VER- UND ENTSchlÜSSELUNG



# FRAGEN?

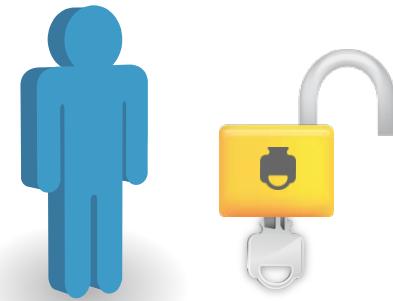
90

# DIAGRAMM



# GEDANKENMODELL ZU ASYMMETRISCHEN CHIFFREN

- Die Idee asymmetrischer Verschlüsselung lässt sich wie folgt veranschaulichen:



Jeder Teilnehmer hat ein Schloss mit passendem Schlüssel.

# GEDANKENMODELL ZU ASYMMETRISCHEN CHIFFREN

- Die Idee asymmetrischer Verschlüsselung lässt sich wie folgt veranschaulichen:

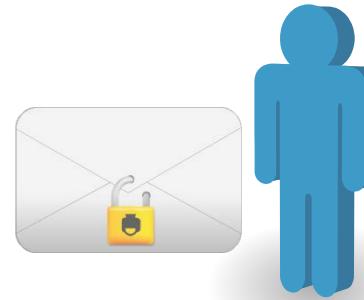


Die Idee ist, Schloss und Schlüssel voneinander zu trennen und Kopien des Schlosses zu veröffentlichen, hingegen den Schlüssel geheim zu halten.

# GEDANKENMODELL ZU ASYMMETRISCHEN CHIFFREN

- Die Idee asymmetrischer Verschlüsselung lässt sich wie folgt veranschaulichen:

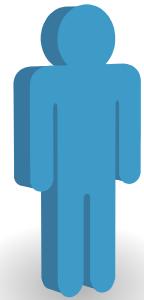
92



Möchte man jetzt jemanden etwas schicken,  
nimmt man dessen Schloss und verschließt die Nachricht damit.

# GEDANKENMODELL ZU ASYMMETRISCHEN CHIFFREN

- Die Idee asymmetrischer Verschlüsselung lässt sich wie folgt veranschaulichen:



Die Nachricht kann dann öffentlich verschickt werden, denn nur der richtige Empfänger kann mit dem passenden Schlüssel das Schloss wieder öffnen.

# ZUGRUNDELIEGENDES PROBLEM

- RSA ist die elektronische Umsetzung des vorigen Gedankenmodells.
- Benannt nach seinen Erfindern Rivest, Shamir und Adleman
- Zugrundeliegendes math. Problem ist die Zerlegung in Primfaktoren.  
Dabei geht es darum, eine große Zahl als Produkt ihrer Primfaktoren darzustellen.
- Zerlegung schwierig bei Zahlen, die nur aus großen Primfaktoren bestehen.  
Bisher gibt es kein effektives und schnelles Verfahren, um diese großen Primfaktoren zu erhalten. Genau darauf beruht die Sicherheit des RSA-Verfahrens.

3347807169895689878604416984821269081770479498371376856891  
2431388982883793878002287614711652531743087737814467999489

93

\* 3674604366679959042824463379962795263227915816434308764267  
6032283815739666511279233373417143396810270092798736308917

= 1230186684530117755130494958384962720772853569595334792197  
6384592519255732630345373154826850791702612214291346167042  
9214311602221240479274737794080665351419597459856902143413

Bit-Länge: 768      Dezimalstellen: 232



Aktuelle PCs können Zahlen mit etwa 80 Dezimalstellen schnell faktorisieren.  
Daher nutzt man RSA real mit Moduli von mindestens 300 Dezimalstellen.

# WIE FUNKTIONIERT DAS RSA-VERFAHREN

- Um zu verstehen, wie das RSA-Verfahren funktioniert, benötigt man einige mathematische Grundlagen.

1 Der Modulo-Operator

2 Eulersche  $\varphi$ -Funktion

3 Satz von Euler/Fermat

# MATHEMATISCHE GRUNDLAGEN - 1

## ■ Der Modulo-Operator

- Dieses Zeichen stellt den Modulo-Operator dar. Beim Modulo-Rechnen betrachtet man den Rest der ganzzahligen Division. D.h. man interessiert sich nur für den Rest, der beim Teilen entsteht, wenn man keine Nachkommastellen zulässt.
- Um es besser zu verstehen, folgendes Beispiel

≡

95



Man möchte sich zu fünf einen Kuchen teilen, der in 16 Stücke aufgeteilt ist.

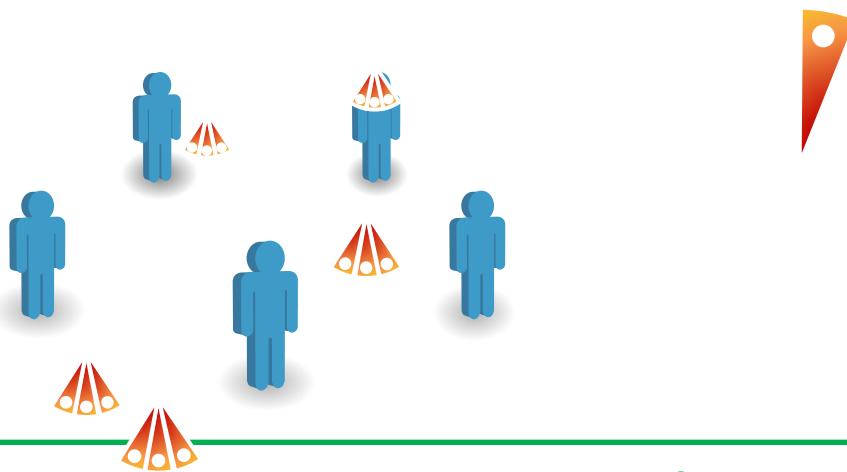
# MATHEMATISCHE GRUNDLAGEN - 1

## ■ Der Modulo-Operator

- Dieses Zeichen stellt den Modulo-Operator dar. Beim Modulo-Rechnen betrachtet man den Rest der ganzzahligen Division. D.h. man interessiert sich nur für den Rest, der beim Teilen entsteht, wenn man keine Nachkommastellen zulässt.
- Um es besser zu verstehen, folgendes Beispiel

≡

95



Jeder kann somit drei Stücke essen. Ein Stück bleibt übrig.

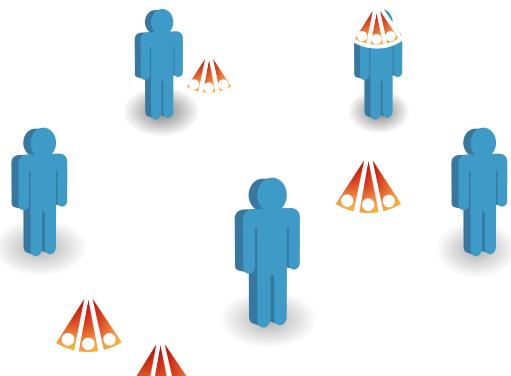
# MATHEMATISCHE GRUNDLAGEN - 1

## ■ Der Modulo-Operator

- Dieses Zeichen stellt den Modulo-Operator dar. Beim Modulo-Rechnen betrachtet man den Rest der ganzzahligen Division. D.h. man interessiert sich nur für den Rest, der beim Teilen entsteht, wenn man keine Nachkommastellen zulässt.
- Um es besser zu verstehen, folgendes Beispiel

≡

95

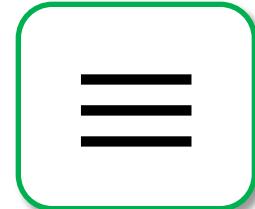


$$16 \equiv 1 \bmod 5$$

Genau diesen Rest berechnet der Modulo-Operator.

## ■ Der Modulo-Operator

- Dieses Zeichen stellt den Modulo-Operator dar. Beim Modulo-Rechnen betrachtet man den Rest der ganzzahligen Division.  
D.h. man interessiert sich nur für den Rest, der beim Teilen entsteht, wenn man keine Nachkommastellen zulässt.



### Mathematische Definition

$$a \equiv b \bmod N$$

bedeutet, dass es eine ganze Zahl  $k$  gibt, so dass sich  $a$  darstellen lässt als

$$a = k * N + b$$

Wobei für  $b$  gelten muss:  $0 \leq b \leq N - 1$



Die Zahl  $k$  ist hierbei uninteressant.  
Wichtig ist nur, dass sie existiert.

### Ein Beispiel für Modulo-Rechnen

96

Der Modulo-Operator ist mit den gewöhnlichen arithmetischen Operationen kommutierbar. Konkret heißt dies, dass es egal ist, ob man erst z.B. eine Multiplikation ausführt,

$$18 * 13 = 234 \equiv 4 \bmod 10$$

oder aber erst modulo rechnet und dann multipliziert:

$$\begin{aligned} 18 * 13 &\equiv 8 * 3 \bmod 10 \\ &= 24 \bmod 10 \equiv 4 \bmod 10 \end{aligned}$$

# MATHEMATISCHE GRUNDLAGEN - 2

## ■ Eulersche $\varphi$ - Funktion

- Die eulersche  $\varphi$  - Funktion einer Zahl  $N$  gibt an, wie viele natürliche Zahlen es gibt, die kleiner als  $N$  und teilerfremd zu  $N$  sind.
- Als Formel sieht dies so aus:

$$\varphi(N) = \#\{a \in \mathbb{N} \mid ggT(a, N) = 1 \text{ und } 1 \leq a < N\}$$



Phi von  $N$  ist die Anzahl derjenigen natürlichen Zahlen  $a$ , für die gilt:  $ggT(a, N) = 1$  und  $1 \leq a < N$

### Wichtige Eigenschaften der $\varphi$ Funktion

Für eine Zahl, die Produkt aus zwei Zahlen  $a$  und  $b$  ist, gilt:

$$\varphi(a * b) = \varphi(a) * \varphi(b)$$

Für Primzahlen  $p$  gilt:

$$\varphi(p) = p - 1$$

Für eine aus zwei Primzahlen zusammengesetzte Zahl  $N = p * q$  gilt somit:

$$\varphi(N) = \varphi(p * q) = \varphi(p) * \varphi(q) = (p - 1)(q - 1)$$

### Beispiel

Wir wollen  $\varphi(10)$  berechnen:

Zunächst faktorisieren wir die Zahl

$$10 = 5 * 2$$

Da die Faktoren Primzahlen sind, können wir nun die Formel links verwenden:

$$\varphi(10) = \varphi(5) * \varphi(2) = 4 * 1 = 4$$

$$\varphi(5) = \#\{1, 2, 3, 4\} = 4 \quad \varphi(2) = \#\{1\} = 1$$

$$\varphi(10) = \#\{1, 3, 7, 9\} = 4$$

## ■ Satz von Euler/Fermat

Für zwei teilerfremde Zahlen  $a$  und  $N$  gilt:

$$a^{\varphi(N)} \equiv 1 \pmod{N}$$

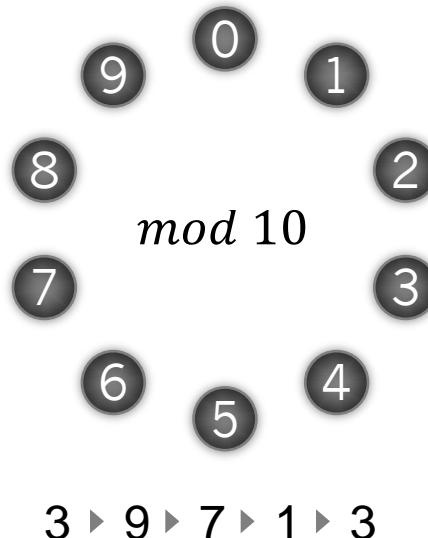


98

Die Ergebnisse der Modulo-  $N$ -Operation sind immer Zahlen aus der **endlichen** Menge  $\{0, 1, \dots, N - 1\}$ .

Funktionen nennt man **zyklisch**, wenn sich die Ergebnisse bei wiederholter Anwendung wiederholen.

Eine solche zyklische Funktion ist z.B. das Potenzieren mit fester Basis: Wir nehmen als Basis die Zahlen  $a = 3$  und  $a = 7$  und multiplizieren sie solange mit sich selbst, bis wir wieder bei der Zahl selbst landen. In unserem Beispiel ist  $N = 10$  mit  $\varphi(N) = 4$ .



Die zwei Zyklen, die dabei entstehen, haben nun beide die Länge 4 , was genau  $\varphi(N)$  entspricht.

# MATHEMATISCHE GRUNDLAGEN - 3

## ■ Satz von Euler/Fermat

Für zwei teilerfremde Zahlen  $a$  und  $N$  gilt:

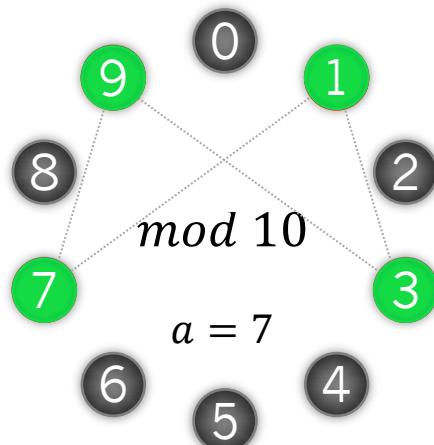
$$a^{\varphi(N)} \equiv 1 \pmod{N}$$



Die Ergebnisse der Modulo-  $N$ -Operation sind immer Zahlen aus der **endlichen** Menge  $\{0, 1, \dots, N - 1\}$ .

Funktionen nennt man **zyklisch**, wenn sich die Ergebnisse bei wiederholter Anwendung wiederholen.

Eine solche zyklische Funktion ist z.B. das Potenzieren mit fester Basis: Wir nehmen als Basis die Zahlen  $a = 3$  und  $a = 7$  und multiplizieren sie solange mit sich selbst, bis wir wieder bei der Zahl selbst landen. In unserem Beispiel ist  $N = 10$  mit  $\varphi(N) = 4$ .



3 → 9 → 7 → 1 → 3

7 → 9 → 3 → 1 → 7

Die zwei Zyklen, die dabei entstehen, haben nun beide die Länge 4, was genau  $\varphi(N)$  entspricht.

Multipliziert man eine solche Zahl  $a$  mit sich selbst, erreicht man in jedem Fall nach  $\varphi(N)$  Multiplikationen wieder die Zahl  $a$ .

Mit diesen Grundlagen können wir uns nun dem eigentlichen Verfahren zuwenden.

# SCHRITT 1: GENERIEREN DER SCHLÜSSEL

- In Schritt 1 generieren wir ein RSA-Schlüsselpaar. Dieser Schritt ist nur einmal initial notwendig.

## Formal

- Wähle zwei Primzahlen  $p$  und  $q$  mit  $p \neq q$
- Bilde ihr Produkt  $N = p * q$
- Berechne den Wert der eulerschen  $\varphi$ -Funktion von  $N$   
$$\varphi(N) = \varphi(p * q) = (p - 1)(q - 1)$$
- Wähle eine Zahl  $e$ , die zwischen 1 und  $N - 1$  liegt und teilerfremd zu  $\varphi(N)$  ist.
- Finde eine weitere Zahl  $d$ , für die gilt:  
$$d * e \equiv 1 \pmod{\varphi(N)}$$

## Am Beispiel

- Wir wählen uns  $p = 13$  und  $q = 7$
  - Damit ist  $N = 13 * 7 = 91$
  - $\varphi(91) = \varphi(13 * 7) = (13 - 1)(7 - 1) = 72$
  - Wir wählen  $e = 5$ , denn es gilt:  
$$\text{ggT}(5, 72) = 1$$
  - Wir nehmen  $d = 29$ , denn dann gilt:  
$$d * e = 145 = 2 * 72 + 1 \equiv 1 \pmod{72}$$
- ?** Hier können Sie erfahren, wie sich eine solche Zahl  $d$  finden lässt.  
(erweiterter euklidischer Algorithmus)

$(e, N)$  ist der *öffentlichen RSA-Schlüssel*.  
 $(d, N)$  ist der *private RSA-Schlüssel*.

# SCHRITT 2: VERSCHLÜSSELN VON NACHRICHTEN

- Nun hat man die Voraussetzungen, um Nachrichten verschlüsselt zu versenden.
- Zuerst muss man die Buchstaben umwandeln, damit man mit ihnen auch rechnen kann.

Dazu kann man zum Beispiel folgende Ersetzung (Substitution) durchführen:

A	B	C	D	...	Z
01	02	03	04	...	26

## Formal

Zum Verschlüsseln der Nachricht, muss nun

$$C = K^e \text{ mod } N$$

gerechnet werden, wobei  $K$  der als Zahl codierte Klartext ist, und  $C$  die verschlüsselte Nachricht (das Chiffrat) darstellt. Die Zahlen  $e$  und  $N$  stammen aus dem öffentlichen RSA-Schlüssel ( $e, N$ ).

Das hier dargestellte Verfahren ist  
deutlich vereinfacht.  
Näheres auf den nachfolgenden Folien.

## Beispiel

Wir führen unser Beispiel nun fort, wählen uns das Wort „GEHEIM“ und verschlüsseln es:

Buchstaben	G	E	H	E	I	M
in Zahlen	07	05	08	05	09	13

Nun wird  $G = 7$  mit Hilfe der Formel links verschlüsselt.  
Unser öffentlicher Schlüssel ist:  $(5, 91)$

$$K^e = 7^5 = 7 * (7^2)^2 = 7 * (49)^2 \equiv 7 * 35 \equiv 63 \text{ mod } 91$$

So wird „GEHEIM“ verschlüsselt in die Zahlen:

63 31 08 31 81 13

# SCHRITT 3: ENTSCHLÜSSELN VON NACHRICHTEN

- Der Empfänger erhält die verschlüsselte Nachricht.

## Formal

Um die erhaltene, verschlüsselte Nachricht zu entschlüsseln, muss der Empfänger rechnen:

$$K = C^d \bmod N$$

Hierbei wird  $K$  den Klartext ergeben. Die Werte  $d$  und  $N$  entnimmt der Empfänger aus seinem privaten Schlüssel  $(d, N)$ .

## Beispiel

Die verschlüsselte Nachricht lautet:

63 31 08 31 81 13

Der Empfänger setzt in die Formel auf der linken Seite 101 seinen geheimen Schlüssel  $(29, 91)$  ein:

$$C^d = 63^{29} = \dots \equiv 7 \bmod 91$$

Nach der Rechnung erhält er wieder den Klartext:

<i>in Zahlen</i>	07	05	08	05	09	13
<i>Buchstaben</i>	G	E	H	E	I	M



Wieso erhält man mit diesen Formeln am Ende wieder den Klartext?  
Eine Erklärung folgt auf der nächsten Folie.

# WAS BEIM VER- UND ENTSchlÜSSELN GESCHIEHT



102

- Erklären lässt sich dies durch die Betrachtung folgender Formeln.
- Wir betrachten den Vorgang des Entschlüsselns der verschlüsselten Nachricht  $C$  genauer:

$$C^d = (K^e)^d = K^{e*d} \bmod N, \text{ da } C = K^e \text{ (Verschlüsselung durch den Absender)}$$

- Es gilt  $d * e \equiv 1 \bmod \varphi(N)$ , was man auch als  $d * e = 1 + l * \varphi(N)$  auffassen kann, wobei  $l$  hier eine beliebige ganze Zahl ist.
- Damit gilt folgende Gleichungskette:

$$K^{e*d} = K^{1+l*\varphi(N)} = K * K^{l*\varphi(N)} = K * (K^{\varphi(N)})^l \bmod N$$

- Benutzt man nun den Satz von Euler/Fermat,  $K^{\varphi(N)} \equiv 1 \bmod N$ , gilt:

$$K * (K^{\varphi(N)})^l \equiv K \bmod N$$

- Insgesamt folgt also, dass gilt:

$$C^d \equiv K \bmod N$$

Beim Potenzieren des Chiffrates mit dem privaten Schlüssel erhält man also wieder den Klartext.



# ZUSAMMENHANG PRIMFAKTOZRZERLEGUNG – RSA-VERFAHREN

- Die Sicherheit des Verfahrens beruht auf dem schwierigen mathematischen Problem der Primfaktorzerlegung.
- Beispiel zur Erklärung:  
Für unsere hier gewählte Zahl  $N$  lässt sich die Primfaktorzerlegung leicht finden:

$$N = 91 = 13 * 7 = p * q$$

- Somit lässt sich ebenfalls  $\varphi(N)$  berechnen. Mit Hilfe des bekannten öffentlichen Schlüssels  $e$  und  $\varphi(N)$  kann man nun den privaten Schlüssel  $d$  finden, da stets  $d * e \equiv 1 \pmod{\varphi(N)}$  gelten muss. Hat man einmal den privaten Schlüssel gefunden, kann man die Nachricht entschlüsseln.
- Es hat noch niemand einen anderen Weg gefunden,
  - um aus dem öffentlichen Schlüssel das  $d$  zu berechnen, außer über die Primfaktorzerlegung.
  - um den Klartext aus dem Chiffrat ohne  $d$  zu berechnen.

103



Die Primfaktorzerlegung erlaubt also, aus dem öffentlichen Schlüssel ( $e, N$ ) den privaten Schlüssel zu berechnen. Der Angreifer führt dabei nach der Primfaktorzerlegung nochmal den initialen Schritt 1 durch.

# FRAGEN?



104

# ZUSAMMENGEFASST

## RSA:

- Verschlüsselung und Entschlüsselung sind Funktionen auf dem Ganzzahl-Ring  $\mathbb{Z}_n$ , wobei  $N=p*q$  das RSA-Modul ist
- RSA verschlüsselt Klartext  $x$ , der ein Element aus  $\mathbb{Z}_n$  sein muss, also ein Element aus  $\{0, 1, \dots, n-1\}$
- Der Zahlwert von  $x$  muss kleiner  $N$  sein,  $x < N$

104

## RSA-Verschlüsselung:

- Gegeben sei Klartext  $x$  und der öffentlicher Schlüssel  $K_E=(e, N)$
- Verschlüsselung  $E(x, K_E) = x^e \text{ mod } N = y$ , mit  $x, y \in \mathbb{Z}_n$

## RSA-Entschlüsselung:

- Gegeben sei der Chiffertext  $y$  und der private Schlüssel  $d = K_D$
- Entschlüsselung:  $x = D(y, K_D) = y^d \text{ mod } N$

# RSA BLOCKGRÖÙE

- Verschlüsselung erfolgt blockbasiert:  
Nachrichtenblock  $x < \text{Modul } N$
- D.h. heißt Größe der zu verschlüsselnden Blöcke abhängig von  $N$
- Wenn  $N$  „zu klein“ werden kleinere Blöcke gewählt so dass  
Zahlendarstellung der Blöcke  $< N$

105

# RSA-VERFAHREN - ANGRIFFE

## ■ Faktorisierungsangriff

- Gegeben: N
- Gesucht: Primfaktoren p und q mit  $p * q = N$
- Ein Verfahren ist die Fermat'sche Faktorisierung
- Führt besonders schnell zum Ziel, wenn p und q sich nahe bei  $\sqrt{N}$  liegen, also wenn p und q ungefähr gleich groß sind
- Konsequenz erfolgreicher Faktorisierung
  - Angreifer kennt Modul N, öffentlichen Schlüssel e und Chiffertext C, aber nicht  $\varphi(N)$
  - Nach Faktorisierung kann  $\varphi(N) = (p - 1) * (q - 1)$  berechnet werden
  - Damit kann er  $d^{-1} = e \text{ mod } \varphi(N)$  und damit den Klartext  $M = C^d \text{ mod } N$  berechnen
- Fazit: bei der Schlüsselgenerierung sollte sichergestellt werden, dass sich p und q um einige Ziffern in ihrer Länge unterscheiden

106

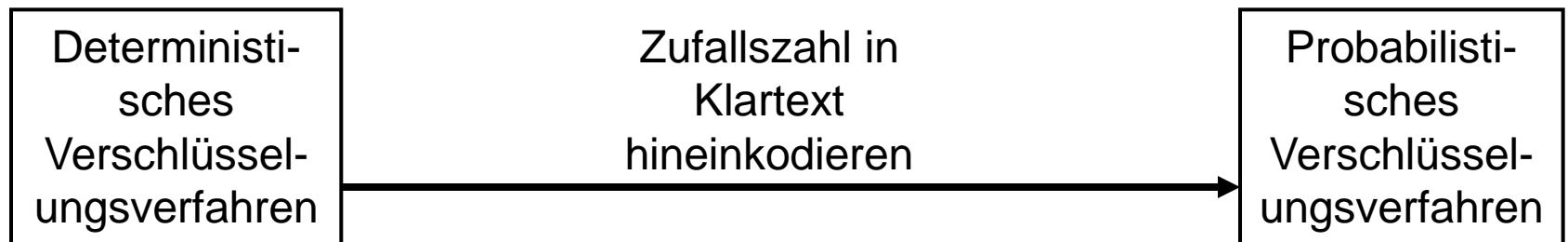
# DURCHGEFÜHRTE FAKTORISIERUNGSANGRIFFE – RSA CHALLENGE

- RSA – Challenge: Aufforderung der Firma RSA bestimmte RSA-Module zu faktorisieren – mit Preisgeld
- 1999 gelang es den RSA-155-Modul (512 Bit) zu faktorisieren
  - Genutzte Ressourcen: 1 Cray Supercomputer (224 Stunden verbrauchte CPU Zeit und 3,2 Gbyte genutzter Hauptspeicher); 300 Workstations und PC (insgesamt verbrauchte Rechenzeit 35,7 CPU-Jahre; insgesamt 7,4 Monate benötigt)
- 2003: RSA-576-Modul (174 stellige Dezimalzahl) von Bonner Mathematikern faktorisiert
- 2005: RSA-640-Modul (193 stellige Dezimalzahl) faktorisiert
- 2007: Bonner Wissenschaftler faktorisieren eine 1039-Bit-Zahl (312 Dezimalstellen)
- RSA stellt überraschend die Challenge zu RSA-1024 (309 Dezimalstellen) und RSA-2048 (617 Dezimalstellen) ein
- **Empfohlen:** Modul mit mind. 2048 Bit, für längerfristige Sicherheit 4096 Bit

107

# RSA – VERFAHREN - ANGRIFFE

- Raten von Klartextblöcken
  - Angreifer kann wahrscheinliche Klartextblöcke raten, mit öffentlichem Schlüssel e verschlüsseln und mit abgefangenen Schlüsseltextrnen vergleichen.
  - In anderen Worten: Da Verschlüsselungsschlüssel e öffentlich, kann jeder/Angreifer beliebige Worte verschlüsseln; sich z.B. ein Wörterbuch aus Klartext und Chiffretext erstellen.
- Verhinderung
  - Zufallszahl in Klartext hineinkodieren (Padding mit Zufallszahl)

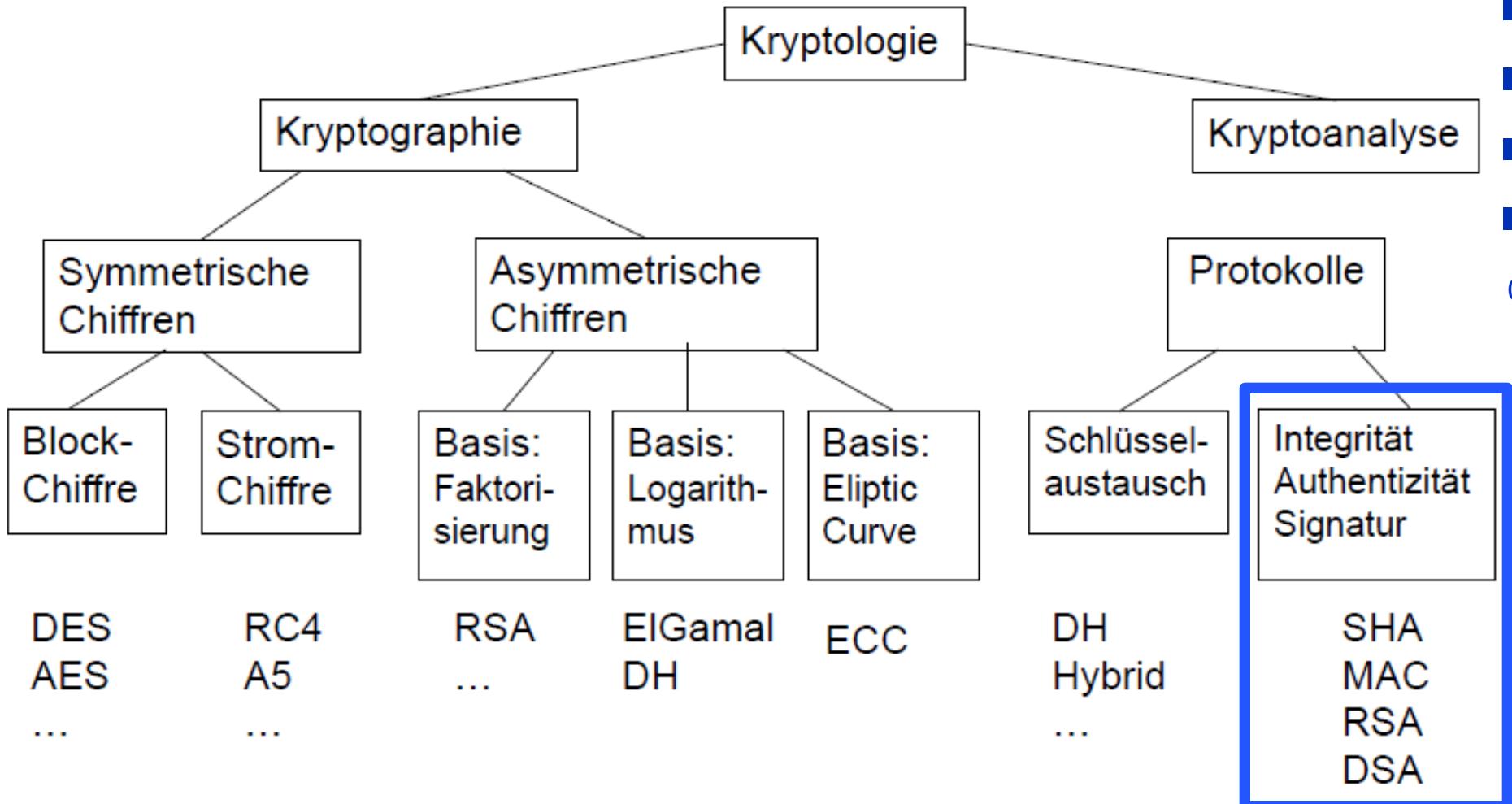


# HYBRIDE VERSCHLÜSSELUNG

- Asymmetrische Verschlüsselung ist im Allgemeinen deutlich langsamer als symmetrische Verschlüsselung
    - Bei RSA liegt dies insbesondere an den zugrundliegenden mathematischen Operationen sowie den sehr großen Zahlen
  - In der Praxis werden deshalb typischerweise symmetrische Verfahren zur Verschlüsselung umfangreicher Daten verwendet und asymmetrische zum Transport des systemmetrischen Schlüssels k
    - Sender erstellt und versendet: RSA-Enc( $e$ ,  $k$ ); AES-Enc( $k$ , Nachricht);
    - Empfänger entschlüsselt zunächst  $k$  mit RSA und dem privaten Schlüssel und dann die Nachricht unter Verwendung von  $k$
- ⇒ Dieses Vorgehen wird als Hybride Verschlüsselung bezeichnet

109

# DIAGRAMM



# KRYPTOGRAPHISCHE HASH-FUNKTIONEN

## Motivation:

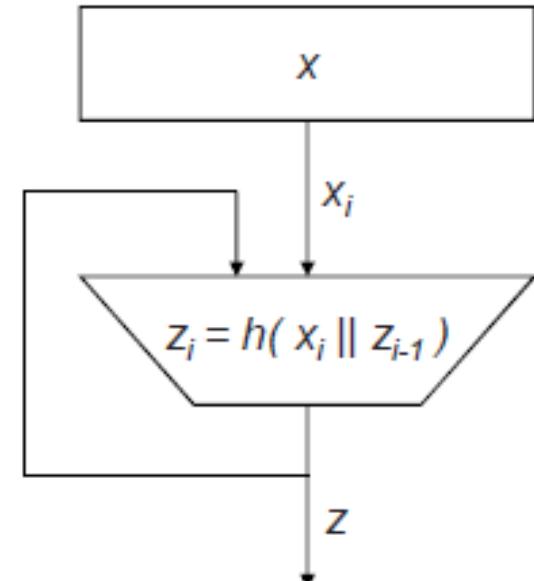
- Integritätsprüfung: Verfahren, um Manipulationen zu erkennen.
  - Sind Prüfsummen a la CRC: geeignet?
- Anforderung an Prüfwert (Message Digest, Fingerprint)?
- Ursprungsnachweise (analog zu Signaturen): Anforderungen?

111

# KRYPTOGRAPHISCHE HASHFUNKTIONEN

## Lösungsansatz:

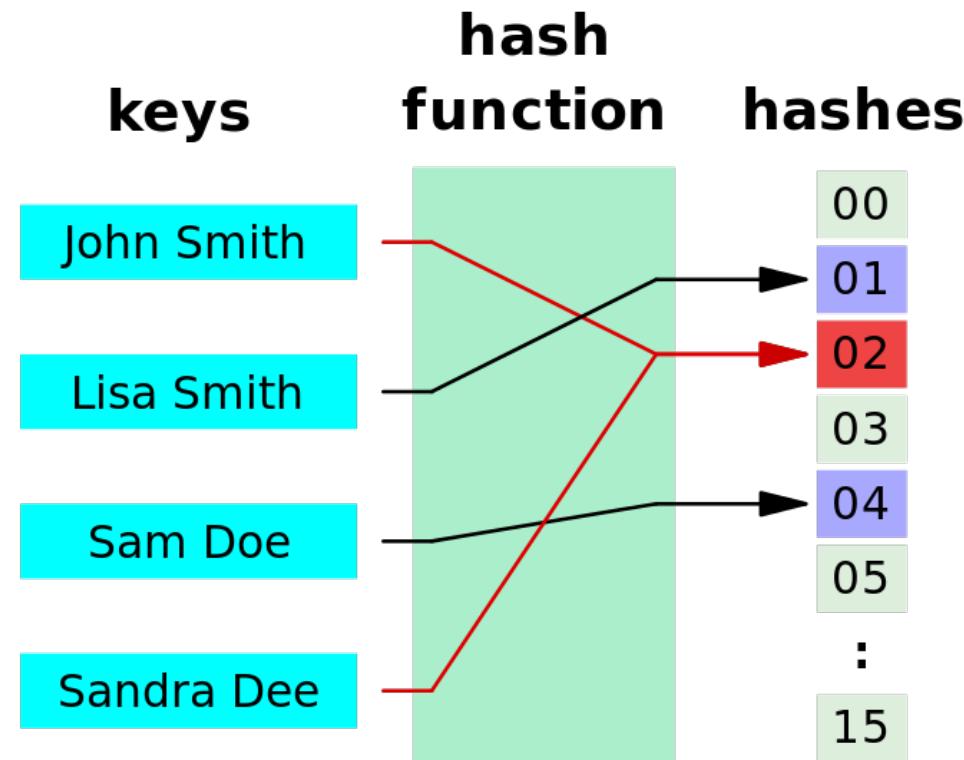
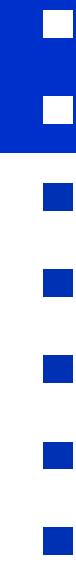
- Nachricht  $x$ , beliebige Länge,  $x_i$  Block
- Kompressionsfunktion  $h$ ,
- kein Schlüsselmaterial notwendig
- $z = h(x)$  kein vertraulicher Wert
- Effizientes Erzeugen eines Message Digest  $z$  fester Länge;  
was ist ausreichende Länge?



112

- $h$  ist eine Abbildung
- $h : X^* \rightarrow X_n$  (Message-Digest, One-way-Hash),  
z.B.  $n = 128$
- $h$  ist eine Hashfunktion, wohlbekannt in der Informatik
- **Aber:**
- $h$  ist nicht injektiv, deshalb sind prinzipiell Kollisionen möglich!

# HASHFUNKTIONEN IN DER INFORMATIK



113

# KRYPTOGRAPHISCHE HASHFUNKTIONEN

Kollision:

- Gegeben seien  $x, y$ , mit  $x \neq y$
- Die Hashfunktion  $H$  produziert eine Kollision, wenn gilt  
 $H(x) = H(y)$

Technik zum Umgang mit Kollisionen

- Hashtabelle, verkettete Listen
- Kollisionen und Integritätsprüfungen: Problem?

114

Folgerung: Kryptographische Hashfunktionen  $H$  notwendig, die spezifische Anforderungen erfüllen

1.  $\forall M \in X^*$  gilt:  $H(M) = y$  ist **einfach** zu berechnen.

2. **Einwegeigenschaft** (Preimage resistance) von H:

- Gegeben  $y = H(M)$ .
- das Bestimmen des Wertes  $M \in X^*$ , mit  $M = H^{-1}(y)$  ist **nicht effizient** möglich

3. **Schwache Kollisionsresistenz** (second Premage)

- Gegeben sei  $M \in X^*$ , es ist **nicht effizient** möglich, ein  $M' \in X^*$  zu finden, so dass gilt:  
 $M \neq M'$  und  $H(M) = H(M')$

4. **Starke Kollisionsresistenz**:

- das Finden von **Paaren**  $M, M' \in X^*$ , mit
- $H(M) = H(M')$  ist **nicht effizient** möglich

# KOLLISIONSRESISTENZ

## Anmerkung:

- Starke Kollisionsresistenz ist schwerer zu garantieren als schwache

## Grund:

- Angreifer hat zwei Freiheitsgrade: beide Nachrichten  $M, M'$  können geändert werden, um gleiche Hashwerte zu finden

Aufwand zum Finden einer Kollision: vgl. Geburtstagsattacke

116

## Geburtstagsattacke:

- Wie viele Personen müssen versammelt sein, damit die Wahrscheinlichkeit, dass 2 Personen den gleichen Geburtstag haben,  $> 0.5$  ist?
- Anzahl der möglichen Werte: 365
- Erste Vermutung: ca. die Hälfte, also 183 Personen sind erforderlich
- Man kann zeigen: schon bei 23 Personen: Wahrscheinlichkeit  $> 0.5$
- Bei 40 Personen: Wahrscheinlichkeit ist  $> 0.9$  !

# GEBURTSTAGSATTACKE

Bedeutung der Geburtstagsattacke für Hashfunktion:

- Gegeben Hashfunktion H, Hashwert der Länge n,  $2^n$  Hashwerte
- Man kann zeigen: die Anzahl der Nachrichten, die man konstruieren muss, um mit einer Wahrscheinlichkeit  $> 0.5$  eine Kollision zu erzeugen liegt bei  
 $\approx \sqrt{2^n}$ , also  $\approx 2^{(n/2)}$

D.h. sei H eine Hashfunktion mit Hashwerten der Länge n = 80

117

- Ein Kollisionsangriff erfordert die Berechnung von  $\approx 2^{40}$  Hashwerten
- Bei heutiger Rechenleistung sind Kollisionsangriffe einfach machbar

Erkenntnis:

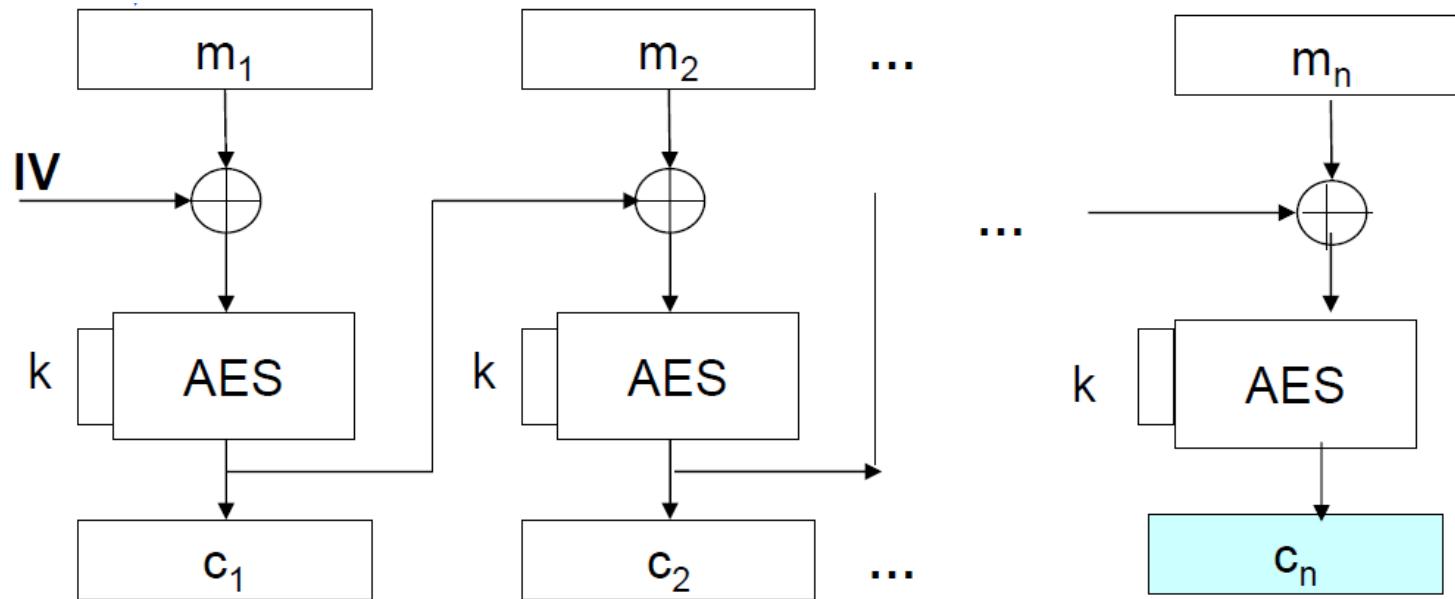
- Länge n ist essentiell, Hashwert sollte mind. 160 Bit lang sein

# HASHFUNKTIONEN

## Klassen von Hashfunktionen:

1. Hashfunktionen basierend auf Block-Chiffren, z.B. DES, AES
2. dedizierte Hashfunktionen: u.a. MD4, MD5 mit 128-Bit Hash oder SHA-1 (Secure Hash-Algorithm), 160-Bit

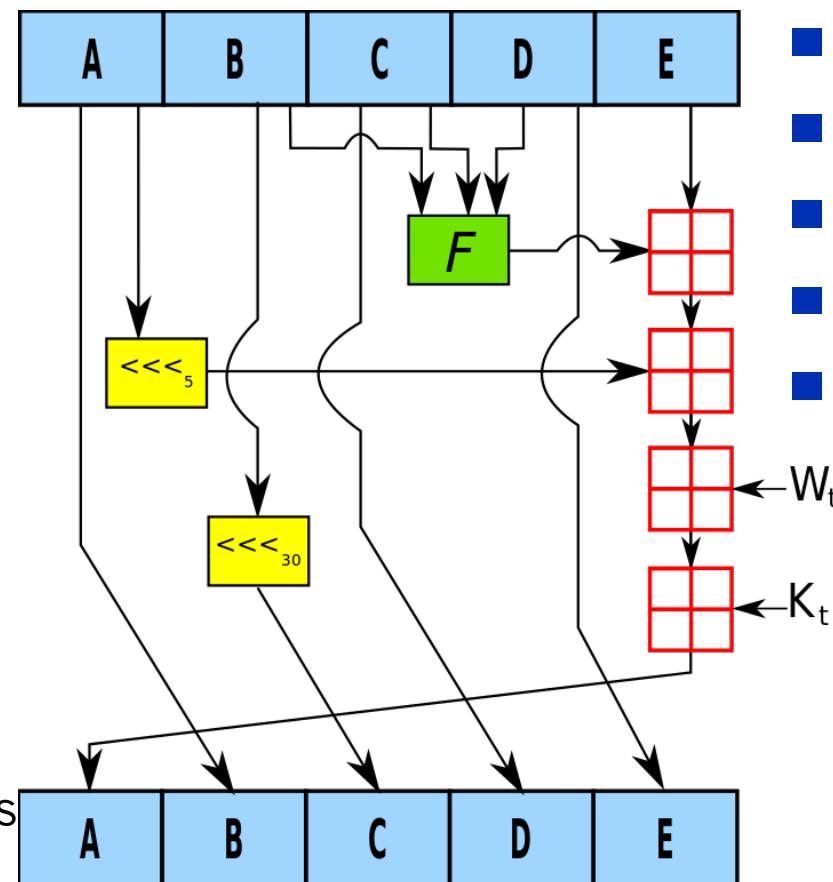
- Beispiel für Block-Chiffren-basierte Hashfunktion
- AES-CBC: der letzte Block (128 Bit) dient als Hash



118

# BEISPIEL FÜR DEDIZIERTE HASHFUNKTION SHA-1 (SECURE HASH-ALGORITHM)

- MD4-basiert,  
Eingabestrings: max  $2^{64}$  Bit lang  
(ca. 2 Exbibyte), Padding
- Verarbeitung von 512-Bit Blöcken,  
Hashwert von 160 bit
- Jeder Block:
  - Verarbeitung in 4 Stufen a 20 Runden
  - Verarbeitung von 32-Bit Worten  $W_i$   
in den 80 Runden, siehe Bild
  - $<<<n$  Linksshift um n-Bits
- Effizient in Software implementierbar:
  - AND, OR, XOR, Komplement und Shifts



Stage t	Round j	Constant $K_t$	Function $f_t$
1	00...19	$K=5A827999$	$f(B,C,D)=(B \wedge C) \vee (\neg B \wedge D)$
2	20...39	$K=6ED9EBA1$	$f(B,C,D)=B \oplus C \oplus D$
3	40...59	$K=8F1BBCDC$	$f(B,C,D)=(B \oplus C) \vee (B \oplus D) \vee (C \oplus D)$
4	60...79	$K=CA62C1D6$	$f(B,C,D)=B \oplus C \oplus D$

# SICHERHEIT DEDIZIERTER HASHFUNKTIONEN

- Das Finden von Kollisionen bei MD4 und dessen Erweiterung MD5 ist möglich! MD5 gilt als unsicher!
- Kollisionen auf SHA-1 (160 Bit) wurden im Jahr 2005 signifikant schneller als Brute Force  $2^{80}$  in  $2^{69}$  herbeigeführt
- Umstieg auf z.B. SHA-2, mit SHA-256 Bit oder SHA-512 Bit wird empfohlen, SHA-2 Verfahren gelten derzeit noch als sicher
- SHA 3 Wettbewerb gestartet in 2007 durch NIST
  - Aus 51 Kandidaten wurde Keccak nach 5 jähriger Prüfung (unter anderem durch die NSA) in 2012 zum neuen Standard erklärt.
  - Hashwerte beliebiger Länge (üblich sind: 224, 256, 384, 512) sehr effizient in Hardware

120

# MESSAGE AUTHENTICATION CODE (MAC)

- Kryptographische Checksummen, Keyed-Hash
- **Ziel:** Authentizität des Datenursprungs und Datenintegrität
- Grundidee:
  - Hashwert zur Überprüfung der Integrität
  - Schlüssel zur Authentifikation des Erstellers des Hashwertes
    - Schlüsselkenntnis erforderlich zur Erstellung eines MAC
- Hashfunktion mit Schlüssel: MAC:  $X^* \times EK \rightarrow X_n$ 
  - geheimer (Pre-shared) Schlüssel  $K_{AB}$  zwischen Partnern A und B
  - $K_{AB}$  symmetrischer Schlüssel (kein Non-Repudiation)
  - MAC-Wert wird an Nachricht angefügt
  - Empfänger prüft Authentizität und Integrität mit  $K_{AB}$
- Ablauf:

121

# BEISPIEL: HASHFUNKTION MIT SCHLÜSSEL

- Beispiel: **Keyed SHA-1**:  $M' = M \mid K_{AB}$  berechnen von  $\text{SHA-1}(M')$ 
  - Beispiel verwendet Secret-Suffix für Schlüssel
  - Auch  $M' = K_{AB} \mid M$  ist möglich, Secret-Präfix
  - **Aber:** Attacken auf beide Varianten sind möglich
- **Angriffsseitee** für Secret-Präfix:  $\text{MAC} = H(K_{AB} \mid M)$ 
  - $M = M_1, \dots, M_n$
  - Angreifer schleust  $M_{n+1}$  ein, ohne den MAC-Schlüssel zu kennen
  - **Vorgehen:**
    - Angreifer fängt  $M, \text{MAC}$  ab
    - Angreifer konstruiert:  $M' = M \mid M_{n+1}$
    - Angreifer konstruiert korrekten  $\text{MAC}' = H(\text{MAC} \mid M_{n+1})$
  - Angriff möglich, da Schlüssel  $K_{AB}$  unverändert Bestandteil der Nachricht bleibt, Empfänger kann Änderung nicht erkennen
  - (Angriffsdetails, siehe Literatur)

122

# HMAC-VERFAHREN

- Vermeidet die Angriffsmöglichkeiten (auf der vorherigen Folie)

## HMAC-Verfahren RFC 2104

- 1996: Bellare, Canetti und Krawczyk
- Idee: HMAC als ‚Wrapper‘ um existierende Verfahren
- HMAC nutzt MAC-Schlüssel, um Initialisierungsvektor zu variieren
- 2-Hash-Anwendungen: innerer und äußerer Hash erhöhen die Kollisionsresistenz

123

### Idee:

- MAC-Schlüssel K nicht nur vorne oder hinten anhängen
- Sondern Schlüssel K, in die Nachricht M hineinziehen‘
- $\text{HMAC} = \text{H}(K \mid (\text{H}(K|M)))$

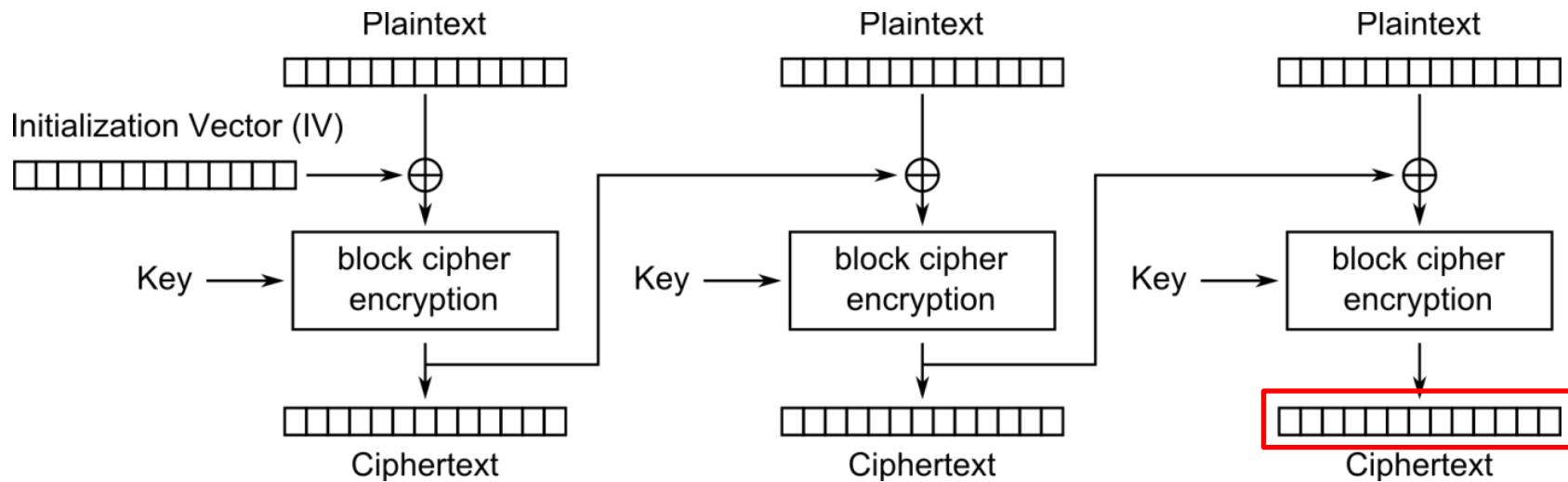
## HMAC: Allgemeine Arbeitsweise

- $h$  ist hierbei eine beliebige kryptographische Hashfunktion, z.B. SHA
- $k$  ist der gemeinsame MAC-Schlüssel
- $x$  ist die Klartext-Eingabe
- $k^+$  ist erweiterter Schlüssel  $k$ : mit 0 aufgefüllt, um Blocklänge des Hashes zu erreichen
- Konstanten
  - $ipad = 00110110, 00110110, \dots, 00110110$
  - $opad = 01011100, 01011100, \dots, 01011100$
- $HMAC_k(x) = h[(k^+ xor opad) || h[(k^+ xor ipad) || x]]$

# BLOCKCHIFFREN-BASIERTE MACs

## ■ CBC-MAC

- Beliebige Blockchiffre z.B. DES oder AES
- Im Blockmodus CBC
- Verschlüsselung der Klartextnachricht mit MAC-Schlüssel  $K_{AB}$
- Verwendung des letzten Chiffre-Textblockes als MAC-Wert



- MAC-Wert = Letzter Chiffretextblock  
hängt von allen Klartextblöcken und Schlüssel ab

# DIGITALE SIGNATUR

**Ziel:** Nachweis der **Urheberschaft** eines Dokuments

**Signaturverfahren:** analog zu Hashfunktionen

- **Dedizierte Signaturverfahren:** z.B. DSA, ECDSA (z.B. im neuen Personalausweis)
  - Hier nicht weiter betrachtet
- **Public Key Verschlüsselung:** z.B. RSA (De-Facto Standard)

126

**Basis:** Schlüsselpaar ( $K_{\text{sig}}$ ,  $K_{\text{veri}}$ ) eines asymmetrischen/Public-Key-Verfahrens

- Privater Signaturschlüssel  $K_{\text{sig}}$
- Öffentlicher Verifikationsschlüssel  $K_{\text{veri}}$

# DIGITALE SIGNATUR MIT RSA

## ■ RSA-Verschlüsselung:

- Gegeben sei Klartext  $x$  und der öffentlicher Schlüssel  $K_E = (e, N)$
- Verschlüsselung  $E(x, K_E) = x^e \text{ mod } N = y$ , mit  $x, y \in \mathbb{Z}_n$

## ■ RSA-Entschlüsselung:

- Gegeben sei der Chiffretext  $y$  und der private Schlüssel  $d = K_D$
- Entschlüsselung:  $x = D(y, K_D) = y^d \text{ mod } N$

- Vertauschen der Ver- und Entschlüsselungsschlüssel

$$(x^e)^d = (x^d)^e$$

- Privater Signaturschlüssel  $K_{\text{sig}}$  (privater Schlüssel  $d$ )
- Öffentlicher Verifikationsschlüssel  $K_{\text{veri}}$  (öffentlicher Schlüssel  $e$ )

- Aufgrund des Rechenaufwandes für RSA werden nicht (lange) Nachrichten signiert sondern (kurze) Hash-Werte
  - Bei RSA sprechen auch Sicherheitsgründe für die Signierung von Hash-Werten anstatt der Nachrichten, mehr dazu gleich
- **Vorgehen:** Signieren eines Klartextes M z.B. mit RSA
  - 1. Hashen
  - 2. Signieren
  - 3. Prüfen

128

# TYPISCHER ABLAUF RSA-SIGNATUR

- Privater Signaturschlüssel  $K_{sig}$  (= privater Schlüssel  $s$ )
- Öffentlicher Verifikationsschlüssel  $K_{veri}$  (öffentlicher Schlüssel  $t$ )
- Bob signiert Nachricht  $m$  an Alice und hängt Signatur an Nachricht an
- Bob signiert mit seinem privaten Signaturschlüssel  $s$

$$\text{sig}_s(m) = (h(m))^s \bmod N$$



- Bob signiert (Hashwert von) Nachricht  $m$  an Alice und hängt Signatur an Nachricht an
- Alice überprüft Bob's Signatur mit dem öffentlichen Schlüssel  $t$  von Bob  
 $h(m)' = \text{sig}_s(m)^t = ((h(m))^s)^t \bmod N$  // Entschlüsseln von  $h(m)$   
 $h(m)^* = h(m)$  // Berechnet Hashwert der empfangenem  $m$   
 $h(m)' =? h(m)^*$  → falls ja OK

# ANGRIFF AUF RSA SIGNATUR

- Nutzen der Multiplikativität von RSA  
(Homomorphismus bzgl. Multiplikation)

(1) Kenntnis von Klartexten  $M_1, M_2$  und deren Signaturen:

- $sig_1 = M_1^s \text{ mod } N$  und  $sig_2 = M_2^s \text{ mod } N$

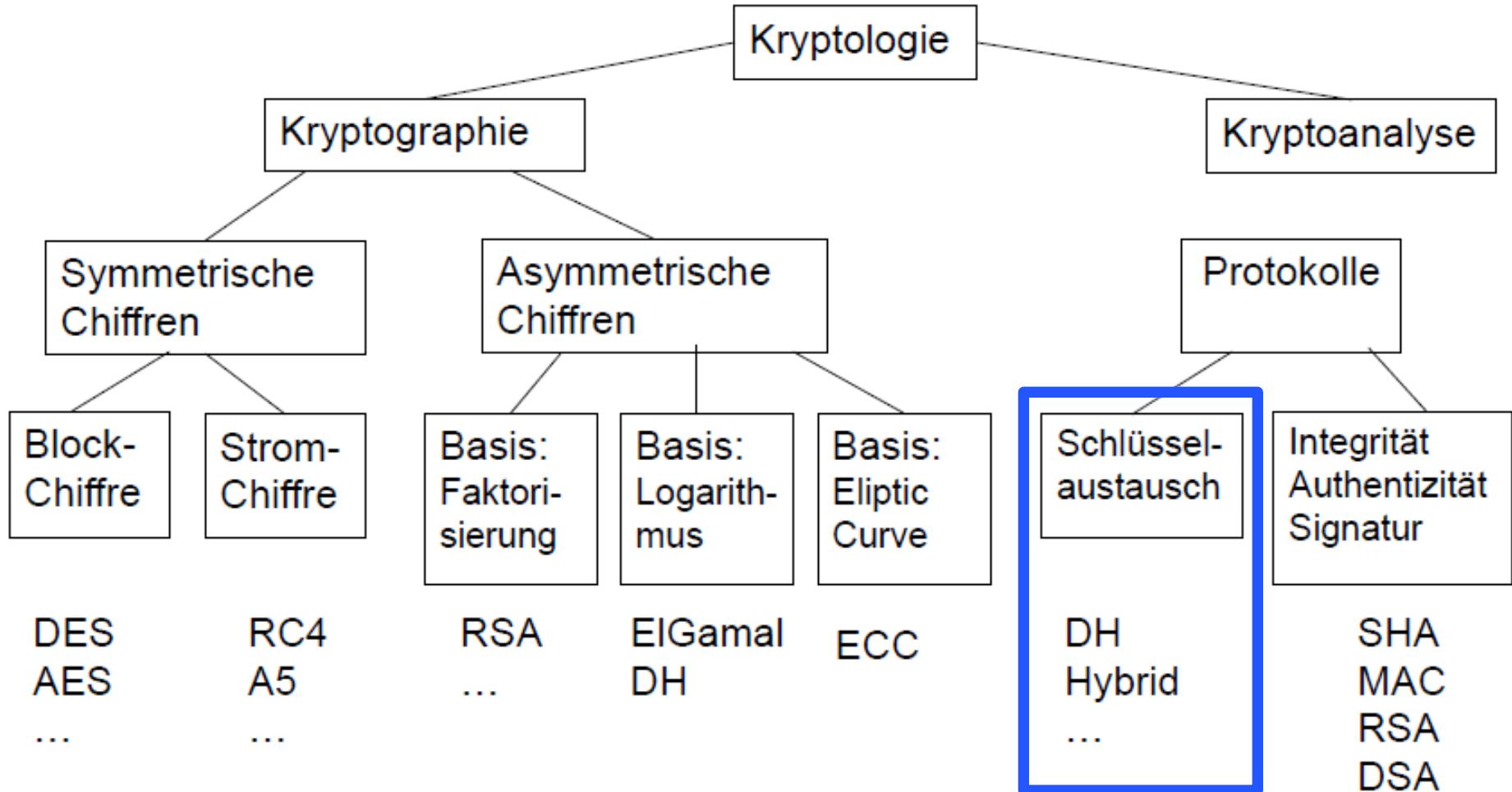
(2) Erzeugen von gültiger  $sig_3$  ohne Kenntnis von  $s$ :

- $(M_1 M_2)^s \text{ mod } N = sig_1 sig_2 \text{ mod } N = sig_3$
- $sig_3$  ist gültige Signatur

- **Lösung:** Hashfunktion  $h$
- Signieren von Hashwerten der Nachrichten
- Warum verhindert das den Angriff?

130

# DIAGRAMM



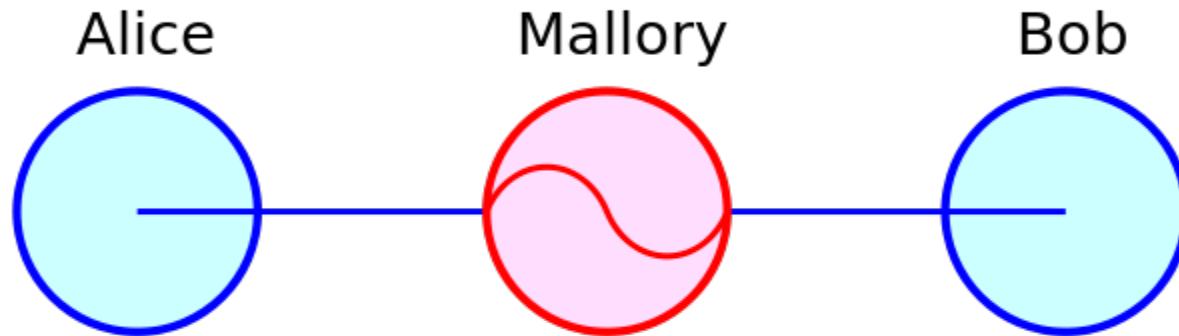
# FRAGEN?



132

# MAN IN THE MIDDLE ANGRIFF GEGEN ASYMMETRISCHE VERFAHREN

- Angreifer hat Vollzugriff auf die Kommunikation zwischen Alice und Bob



133

- Angriff bei allen asymmetrischen Verfahren möglich
- Angreifer Mallory fängt zwischen Alice und Bob ausgetauschte öffentliche Schlüssel ab und ersetzt sie seinen eigenen

# Man in the Middle-Attacke

Mallory



Schlüssel	öffentlich	privat
Mallory	red key icon	red key icon
Alice		
Bob		

Bob



Schlüssel	öffentlich	privat
Bob	green key icon	green key icon
Alice		

Alice



Schlüssel	öffentlich	privat
Alice	blue key icon	blue key icon
Bob		

Alice und Bob beginnen ihre Kommunikation mit dem Austausch ihrer öffentlichen Schlüssel.

Bob sendet seinen Schlüssel zuerst.

Mallory fängt das Datenpaket mit Bob's öffentlichen Schlüssel ab und ersetzt ihn durch seinen eigenen öffentlichen Schlüssel.

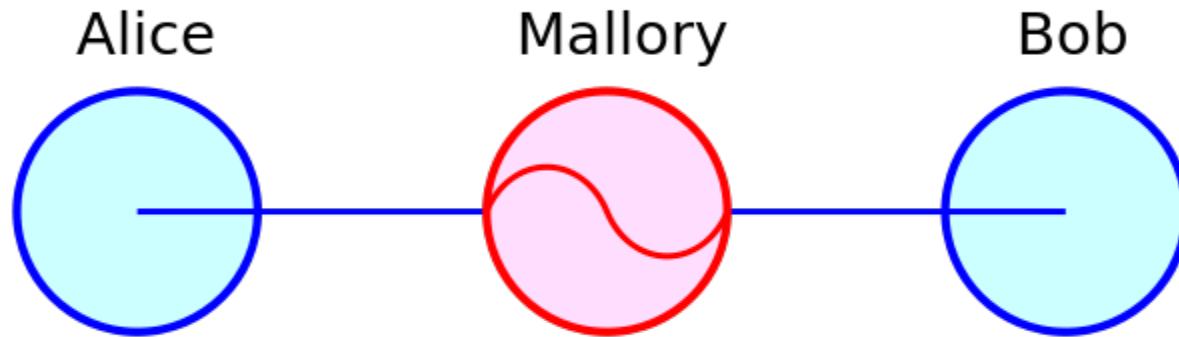
Er speichert Bob's Schlüssel ab.

[zurück](#)

[weiter](#)

# MAN IN THE MIDDLE ANGRIFF GEGEN ASYMMETRISCHE VERFAHREN

- Angreifer hat Vollzugriff auf die Kommunikation zwischen Alice und Bob



135

- Angriff bei allen asymmetrischen Verfahren möglich
- Angreifer Mallory fängt zwischen Alice und Bob ausgetauschte öffentliche Schlüssel ab und ersetzt sie seinen eigenen
- Was ist das zugrundeliegende Problem das eine Man in the Middle Attacke ermöglicht?
- Die öffentlichen Schlüssel sind nicht authentifiziert: Wenn Alice einen öffentlichen Schlüssel vermeintlich von Bob erhält, woher weiß sie, dass er wirklich von Bob?

# Schlüsselmanagement

- Ziele und Rolle
  - Schlüsselbeziehungen, Schlüsselmanagement
- Einfache Schlüsselmanagementmodelle
  - Punkt-zu-Punkt vs. Zentralisiertes
- Trusted Third Parties
  - Rollen; Vertrauen, Vertrauensstufen; Kategorien
- Techniken des Schlüsselmanagements
- Aufgaben des Schlüsselmanagement
- Schlüsseletablierungsprotokolle

136

# Schlüsselmanagement

- **Schlüsselbeziehungen** bestehen zwischen kommunizierenden Parteien, die zur Nutzung kryptographischer Verfahren gemeinsame Daten teilen, z.B. geheime oder öffentliche Schlüssel oder Initialisierungswerte
- **Schlüsselmanagement** umfasst Techniken und Prozeduren zur Etablierung und Aufrechterhaltung von Schlüsselbeziehungen zwischen autorisierten Parteien
- **Aufgabe** des Schlüsselmanagements ist die Wartung von Schlüsselbeziehungen und Schlüsselmaterial auf eine Art und Weise die dem
  - Verlust der Vertraulichkeit geheimer Schlüssel und dem
  - Verlust der Authentizität von geheimen und öffentlichen Schlüssel begegnet.

137

# SCHLÜSSELMANAGEMENT

- Ziel eines guten kryptographischen Entwurfs
  - Reduktion komplexer Probleme auf die Verwaltung einer geringen Zahl kryptographischer Schlüssel
  - Schlüssel letztlich durch vertrauenswürdige Hardware oder Software, physikalische Isolation oder organisatorische Maßnahmen gesichert
  - Abhängigkeit von organisatorischer Sicherheit, vertrauenswürdiger Hardware und dem Vertrauen in eine Vielzahl von Funktionen wird minimiert durch Konzentration des Vertrauens auf eine kleine Zahl leicht zu überwachender und zu steuernder, vertrauenswürdiger Elemente.

138

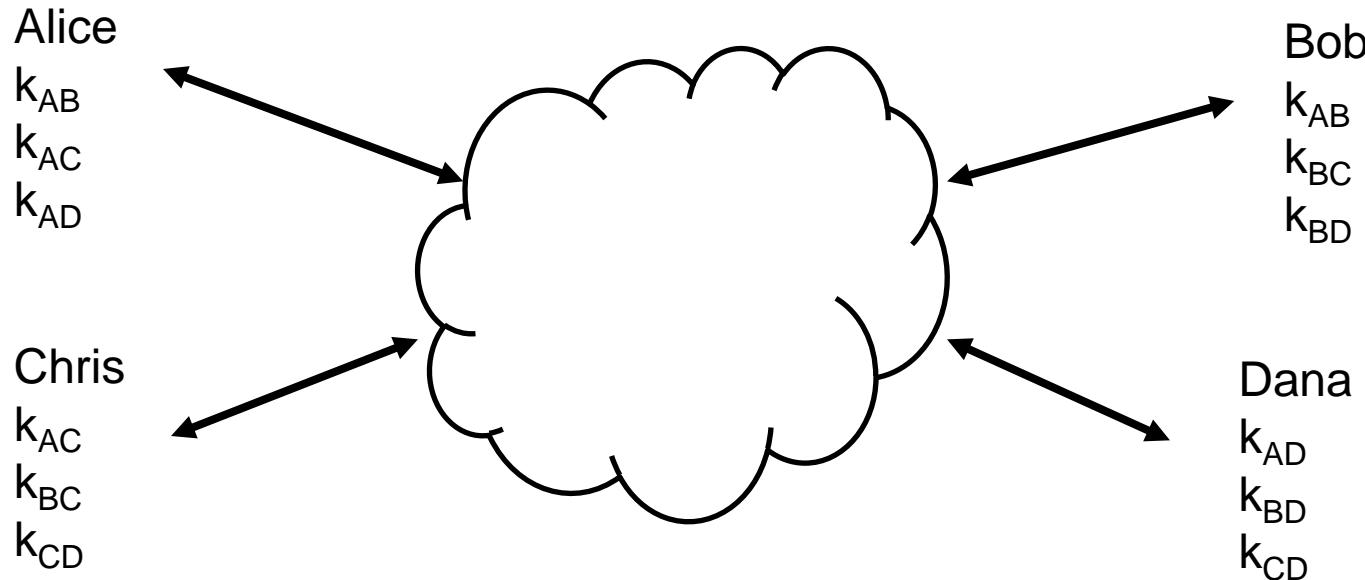
# EINFACHE SCHLÜSSELMANAGEMENTTECHNIKEN

- Punkt-zu-Punkt-Kommunikation
- Zentrale Schlüsselserver bei symmetrischer Kryptographie
  - Schlüsselverteilungszentren
  - Schlüsselübersetzungszentren

139

# EINFACHEN SCHLÜSSELMANAGEMENTMODELLE

- Punkt-zu-Punkt-Kommunikation
  - $n^2$ -Schlüsselverteilungsproblem
  - Situation: Netz mit  $n$  Nutzern, die alle sicher mit allen anderen  $n-1$  Nutzern kommunizieren wollen
  - Einfacher Ansatz: jedes Nutzer-Paar erhält individuellen Schlüssel



140

# EINFACHEN SCHLÜSSELMANAGEMENTMODELLE

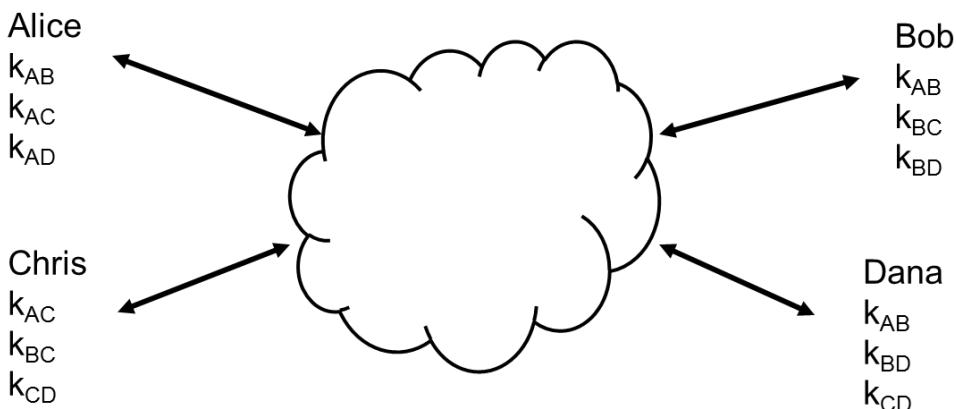
## ■ Punkt-zu-Punkt-Kommunikation

- $n^2$  Schlüsselverteilungsproblem

- Nachteile des Einfachen Ansatzes

- Es gibt  $n(n-1) \approx n^2$  Schlüssel im System
- Es gibt  $n(n-1)/2$  Schlüsselpaare
- Für neuen Nutzer Eddi müssen Schlüssel  $k_{XE}$  sicher zu allen vorhanden Nutzern X transportiert werden
- ⇒ nur für kleine relativ statische Netze realisierbar

141



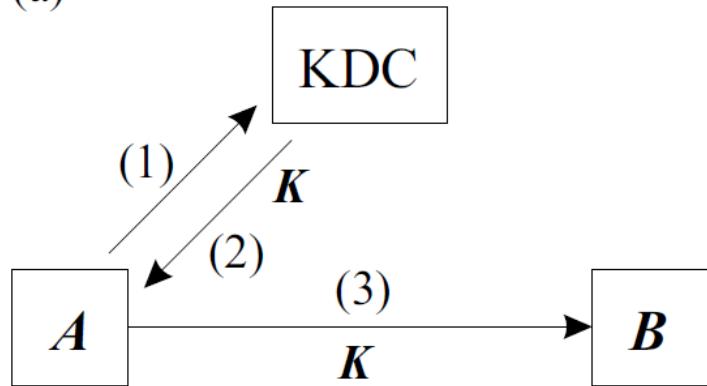
Beispiel: mittelgroße Firma mit 750 Angestellten

- $750 * 749 = 561750$  Schlüssel müssen sicher verteilt werden

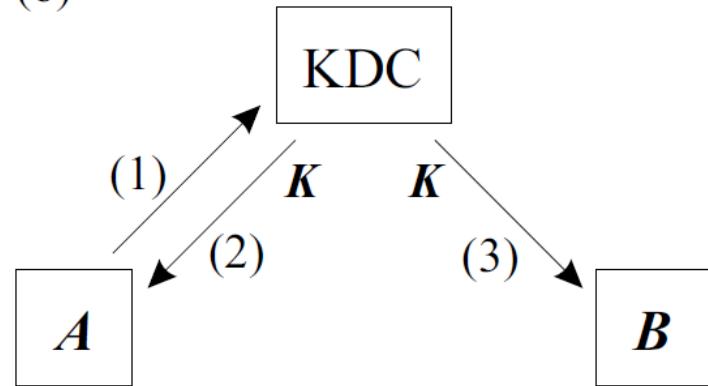
# SCHLÜSSELVERTEILZENTREN

- Zentrale Schlüsselserver bei symmetrischer Kryptographie
  - Schlüsselverteilzentrum T (Key Distribution Center - KDC)
  - Jeder Nutzer x besitzt nur gemeinsamen Schlüssel mit KDC:  $k_{xT}$

(a)



(b)



(1) Anfrage an KDC wegen  $k_{AB}$

(2) KDC T generiert  $k_{AB}$  und sendet diesen verschlüsselt mit  $k_{AT}$  und  $k_{BT}$  an A

(3) A leitet  $k_{AB}$  verschlüsselt mit  $k_{BT}$  an B weiter

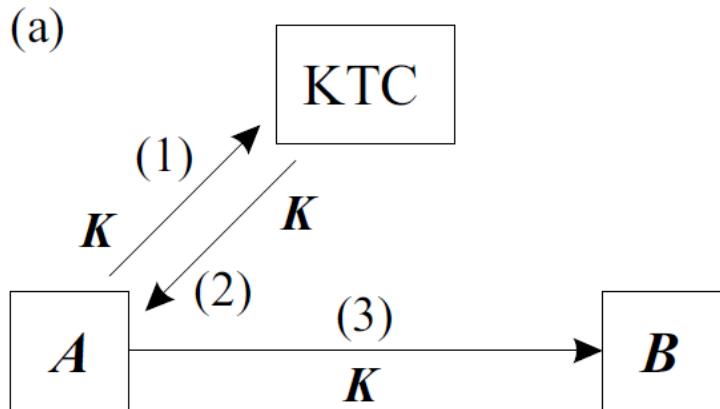
(1) Anfrage an KDC wegen  $k_{AB}$

(2) KDC T generiert  $k_{AB}$  und sendet diesen verschlüsselt mit  $k_{AT}$  an A

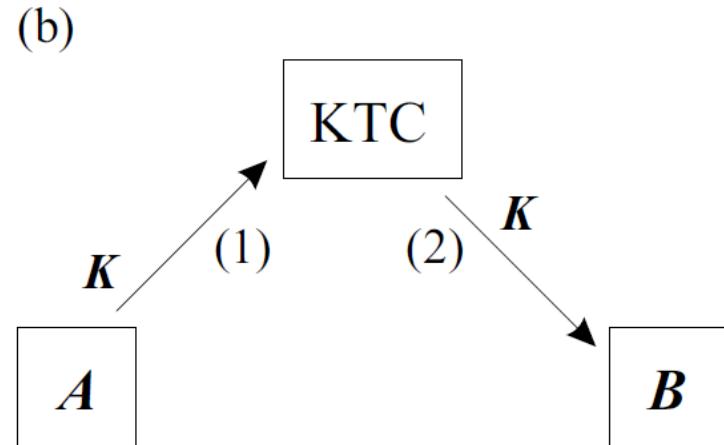
(3) KDC sendet  $k_{AB}$  verschlüsselt mit  $k_{BT}$  direkt an B

# SCHLÜSSELÜBERSETZUNGSZENTREN

- Zentrale Schlüsselserver bei symmetrischer Kryptographie
  - Schlüsselübersetzungszentrum T (Key Translation Center – KTC)
  - Ähnlich KDC aber statt KDC generiert A den Schlüssel  $k_{AB}$



- (1) A sendet Namen B und generierten  $k_{AB}$  verschlüsselt mit  $k_{AT}$  an KTC T  
(2) KTC entschlüsselt  $k_{AB}$  und sendet diesen verschlüsselt mit  $k_{BT}$  an A (zur Weiterleitung an B)  
(3) A leitet weiter an B



- (1) A sendet Namen B und generierten  $k_{AB}$  verschlüsselt mit  $k_{AT}$  an KTC T  
(2) KTC T entschlüsselt  $k_{AB}$  und sendet diesen verschlüsselt mit  $k_{BT}$  direkt an B

# EINFACHEN SCHLÜSSELMANAGEMENTMODELLE

- Punkt-zu-Punkt-Ansätze erfordern dass jeder mit jedem einen gemeinsamen Schlüssel besitzt  $\Rightarrow n^2$  Schlüssel
- Zentralisierte Ansätze (KDC/KTC) erfordern einen vertrauenswürdigen Server T sowie das jede Partei einen gemeinsamen Schlüssel mit T besitzt.
  - (+) Keine  $n^2$  Schlüssel, jede Partei hat nur einen Schlüssel mit T anstatt  $n \cdot 1$  mit allen anderen Parteien
  - (-) kompletter Sicherheitsverlust, wenn (das attraktive Angriffsziel) KDC/KTC T kompromittiert wird
  - (-) KDC/KTC T ist ggf. Performance-Flaschenhals bei starker Auslastung
  - (-) Vertrauenswürdiger Online-Server T erforderlich

144

# VERTRAUENSWÜRDIGE DRITTE PARTEIEN

- Trusted Third Parties (TTPs) übernehmen verschiedene Rollen
  - Symmetrische Krypto: z.B. KDC und KTC
  - Asymmetrische Krypto: Zertifizierungsinstanz (mehr zu Zertifikaten später) bescheinigt die Zugehörigkeit eines öffentlichen Schlüssels zu einer Person/Name
  - Name-Server, Schlüsselgenerierungsdienst, Zeitserver, Schlüsselhinterlegung, ...
- Je nach Rolle der TTP unterschiedliches Maß an Vertrauen und Kompetenz erforderlich, z.B.
  - Verrät keine Geheimnisse
  - Erstellt keine falschen Zertifikate
  - Gibt keine falschen Auskünfte

145

# VERTRAUENSWÜRDIGE DRITTE PARTEIEN

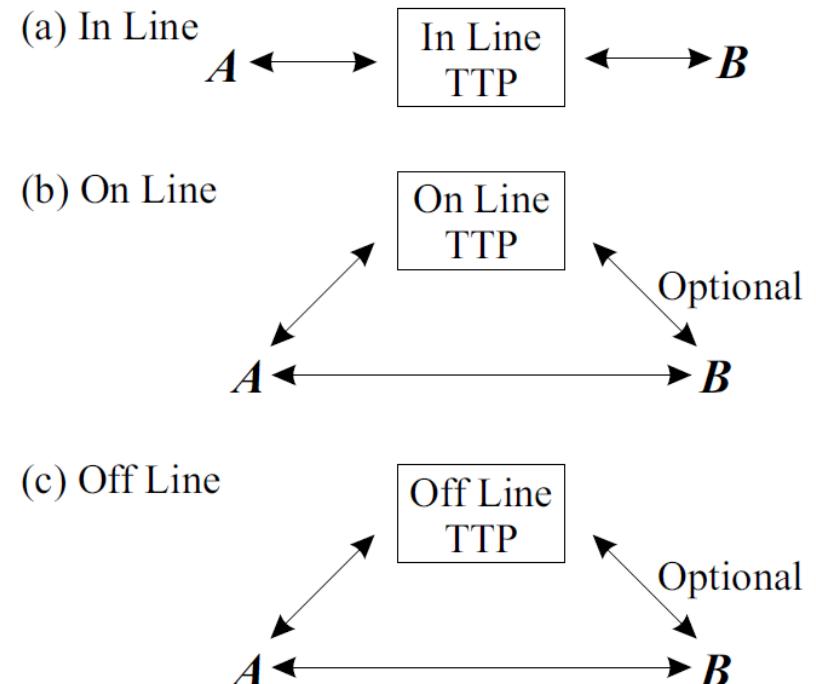
- Im allgemeinen 3 Stufen des Vertrauens in eine TTP T unterschieden
  - Stufe 1: T kennt den geheimen Schlüssel des Benutzers.
  - Stufe 2: T kennt nicht den geheimen Schlüssel der Benutzer, aber T kann falsche Bescheinigungen erstellen ohne dass diese als falsch erkannt werden.
  - Stufe 3: T kennt keine geheimen Schlüssel und gefälschte Bescheinigungen sind erkennbar

146

# VERTRAUENSWÜRDIGE DRITTE PARTEIEN

## ■ Unterscheidung von TTPs anhand Echtzeitinteraktion

- In-Line TTP ist Vermittler, der in Echtzeit an der Kommunikation beteiligt ist
- On-Line TTP T ist in Echtzeit an Kommunikation beteiligt, Kommunikation jedoch nicht **über** T
- Off-Line TTP T ist nur vorbereitend und nicht in Echtzeit an Kommunikation beteiligt
- Widerrufe von z.B. Privilegien bei On-Line und In-Line einfacher realisierbar

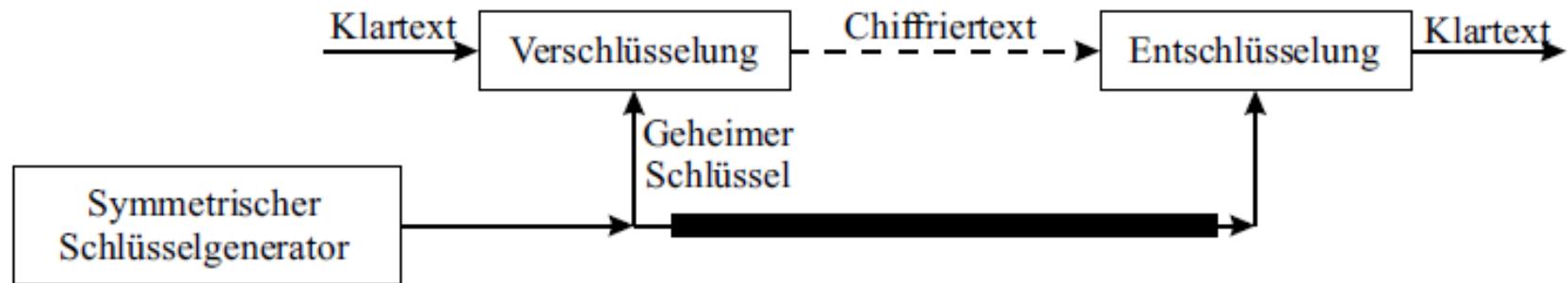


# SCHUTZBEDARFE VON SCHLÜSSELN

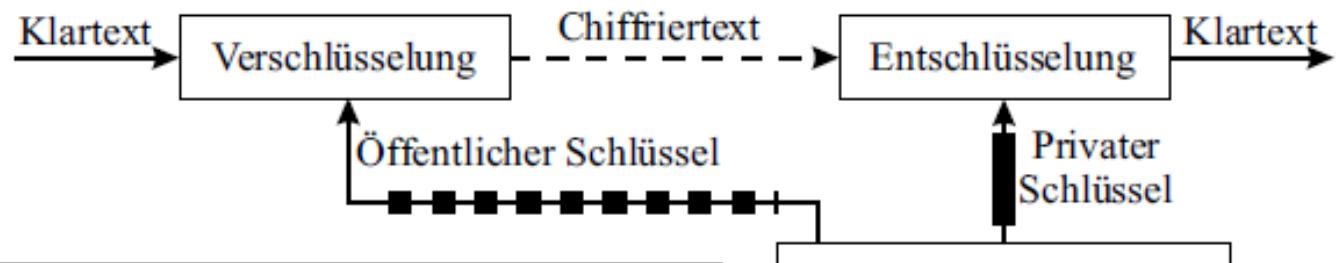
Schlüssel zur	Symmetrische Schlüssel		Asymmetrische Schlüssel	
	Vertraulichkeit	Authentizität	Vertraulichkeit	Authentizität
Verschlüsselung	Ja	Ja	Nein	Ja
Entschlüsselung	Ja	Ja	Ja	Ja

# SCHUTZBEDARF VON SCHLÜSSELN

## (a) Symmetrische Verschlüsselung



## (b) Asymmetrische Verschlüsselung



### Legende:

- sicherer Kanal (Vertraulichkeit und Authentizität)
- - - sicherer Kanal (nur Authentizität)
- - - - unsicherer Kanal

# AUFGABEN DES SCHLÜSSELMANAGEMENTS

- Schlüsselerzeugung
  - Zufallszahlen!
- Übermittlung von Schlüsseln
  - Persönliches Treffen
  - Nutzung eines sicheren Kanals
  - Schlüssel aufteilen und Übermittlung der Teile auf verschiedenen Kanälen, die Angreifer (hoffentlich) nicht alle abhören kann
  - Verifizierung von Schlüsseln
    - Wie kann sich B beim Empfang eines Schlüssels davon überzeugen, dass er tatsächlich von A stammt und nicht von jemand anders der sich als A ausgibt?
    - Unterzeichnet A den Schlüssel digital muss sich B auf die Echtheit des vorliegenden öffentlichen Signaturverifikationsschlüssel verlassen und darauf, dass A seinen privaten Signaturschlüssel sicher/vertraulich verwahrt.
    - Wird A's öffentlicher Schlüssel von einer TTP T unterschrieben/zertifiziert muss sich B darauf verlassen dass der vorliegende öffentliche Schlüssel von T nicht manipuliert wurde.

150

# AUFGABEN DES SCHLÜSSELMANAGEMENTS

## ■ Verwendung von Schlüsseln

- Vertraulichkeitsschutz von Schlüsseln im Hauptspeicher gegen andere Prozesse, Benutzer, Systemadministratoren
  - Was kann man tun?
- Gültigkeitsdauer von Schlüsseln sollte begrenzt sein (wie Reisepass oder Lizenz auch), weil:
  - Je länger ein Schlüssel in Verwendung,
    - um so wahrscheinlicher ist seine Kompromittierung,
    - um so größer ist der Schaden nach einer Kompromittierung und
    - um so attraktiver ist er für einen Angreifer.
  - Kryptoanalyse mit mehr Chiffretext, der mit demselben Schlüssel erstellt wurde, ist generell einfacher.
  - Bestimmte Technologien (hier Kryptoverfahren) sollten nicht länger als ihre geschätzte Lebensdauer eingesetzt werden.

151

# AUFGABEN DES SCHLÜSSELMANAGEMENTS

- Verwendung von Schlüsseln (Forts.)
  - Aktualisierung von Schlüsseln
    - Neu generieren und verteilen
    - Alternativ, Schlüsselableitung:  
Neuer-Schlüssel = Einwegfunktion/Hash(Alter-Schlüssel)
      - Neuer Schlüssel **nicht** sicherer als alter Schlüssel!
      - Wenn alter Schlüssel kompromittiert dann kann auch neuer abgeleitet werden
- Speicherung von Schlüsseln
  - Auf Chipkarten oder ähnlichem
  - Im Kopf
    - Schlüssel die schwer zu merken sind, können mit anderem Schlüssel/Passwort verschlüsselt gespeichert werden und Passwort wird im Kopf gespeichert
  - Sicherheitskopien von Schlüsseln erforderlich
    - Mitarbeiter verlässt Unternehmen; Wer kommt an seine Daten?
    - Hinterlegung von Schlüsselkopien birgt Missbrauchspotential
      - Weitere Lösungen mit Secret Sharing möglich. Grundidee: es gibt 10 Schlüsselteile, beliebige 5 davon sind zum öffnen der Tür erforderlich. (hier nicht näher betrachtet)

152

# AUFGABEN DES SCHLÜSSELMANAGEMENTS

- Reaktion auf die Kompromittierung eines Schlüssels
  - Symmetrischer Schlüssel kompromittiert (nicht mehr geheim)
    - Durch neuen ersetzen
    - Hoffentlich ist der entstandene Schaden gering
  - Asymmetrischer privater Schlüssel kompromittiert (nicht mehr geheim) ⇒ größere Probleme
    - Zugehöriger öffentlicher Schlüssel an vielen Stellen in Verwendung
    - Angreifer kann privaten Schlüssel nutzen (Signieren, Entschlüsseln)
      - ⇒ Nachricht von der Kompromittierung sofort weiterverbreiten
      - ⇒ Alle Datenbanken öffentlicher Schlüssel müssen benachrichtigt werden, damit öffentlicher Schlüssel nicht weiter verwendet wird
- ⇒ Unterstreicht Problematik, wenn ein einziger Schlüssel für alle kryptographischen Anwendungen verwendet wird.
  - ⇒ Besser unterschiedliche Schlüssel für unterschiedliche Anwendungen

153

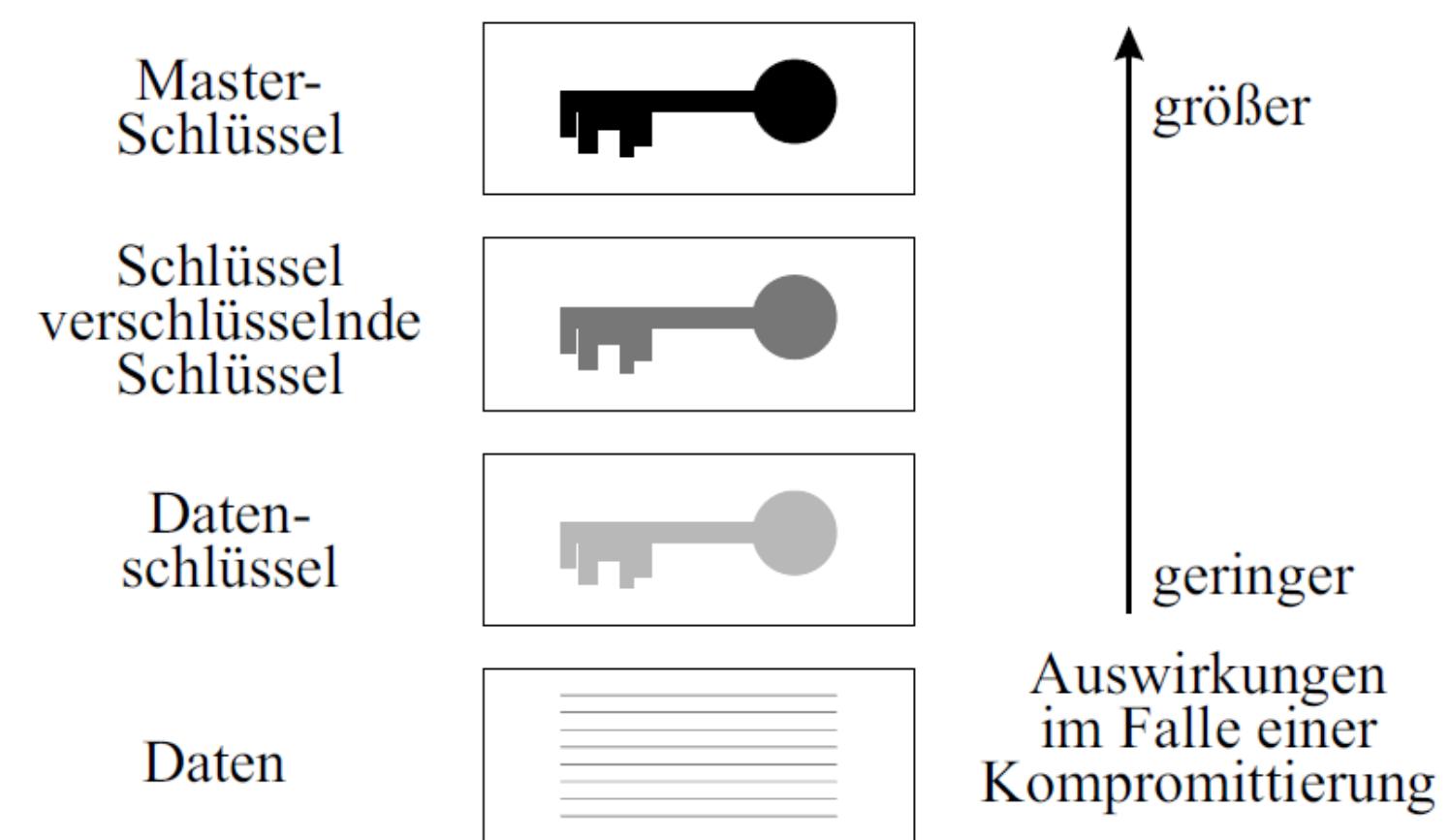
# AUFGABEN DES SCHLÜSSELMANAGEMENTS

- Sicheres Löschen von Schlüsseln
  - Nach Ersetzung durch neue müssen alte Schlüssel gelöscht werden
  - Mit alten Schlüsseln könnten alte Nachrichten entschlüsselt werden
  - Wie löscht man sicher?
    - Peter Gutmann beschreibt in einem Artikel 1996 Verfahren zur Wiederherstellung gelöschter Daten, die auf magnetischen Medien gespeichert wurden.
    - Zum sicheren Löschen dieser Daten empfiehlt er diese 35 mal (achtmal mit zufälligen und 27 mal mit speziellen Werten) zu überschreiben.
    - Wenn Daten wertvoll genug sind, sollte sogar davon ausgegangen werden, dass es unmöglich ist, sie vollständig von magnetischen Medien zu löschen.

154

- Verteilung und Speicherung geheimer Schlüssel
  - **Schlüsselhierarchien**, ausgehend von der Rolle von Schlüsseln
    - Master-Schlüssel
      - werden initial installiert, manuell verteilt und extern (im Kopf oder durch physikalische Isolation) geschützt
    - Schlüssel verschlüsselnde Schlüssel
      - Für Speicherung oder Transport anderer Schlüssel verwendet
      - Aka Schlüsseltransportschlüssel
      - Durch Verschlüsselung mit Masterschlüsseln geschützt
    - Datenschlüssel
      - Für kryptographische Verfahren auf Benutzerdaten
      - Meist symmetrische Schlüssel mit relativ kurzer Lebensdauer
      - Z.B. Sitzungsschlüssel

# Schlüsselhierarchie

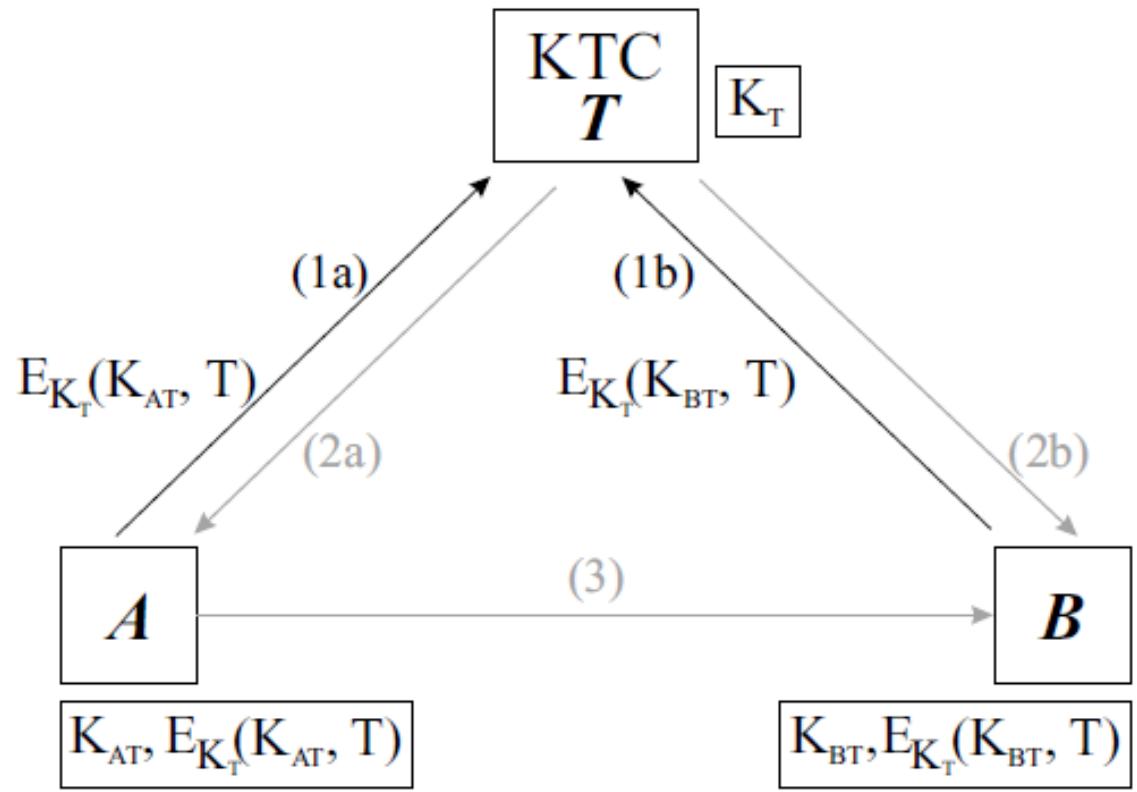


- Schlüssel einer Ebene werden durch Schlüssel höherer Ebenen geschützt
- Erhöhter Schutzbedarf auf höheren Ebenen

- Verteilung und Speicherung geheimer Schlüssel
  - Symmetrische Schlüsselzertifikate
    - Eine KTC T muss für jede Partei A, B, ..., X, jeweils einen gemeinsamen geheimen Schlüssel speichern:  $k_{AT}$ ,  $k_{BT}$ , ...,  $k_{XT}$
    - Durch Verwendung symmetrischer Schlüssel kann diese zentrale Speicherung entfallen
    - KTC T verwendet geheimen symm. Schlüssel  $k_T$  (den nur T kennt) zur Verschlüsselung jedes  $k_{XT}$  und die Speicherung des verschlüsselten  $k_{XT}$  erfolgt bei Partei X
    - Vor/zu Beginn der Kommunikation einer Partei X mit T übermittelt X das verschlüsselte  $k_{XT}$  (das symm. Schlüsselzertifikat) an T

# SYMMETRISCHE SCHLÜSSELZERTIFIKATE

158



- Nach (1a) und (1b)  
wie klassisches KTC

## Legende:

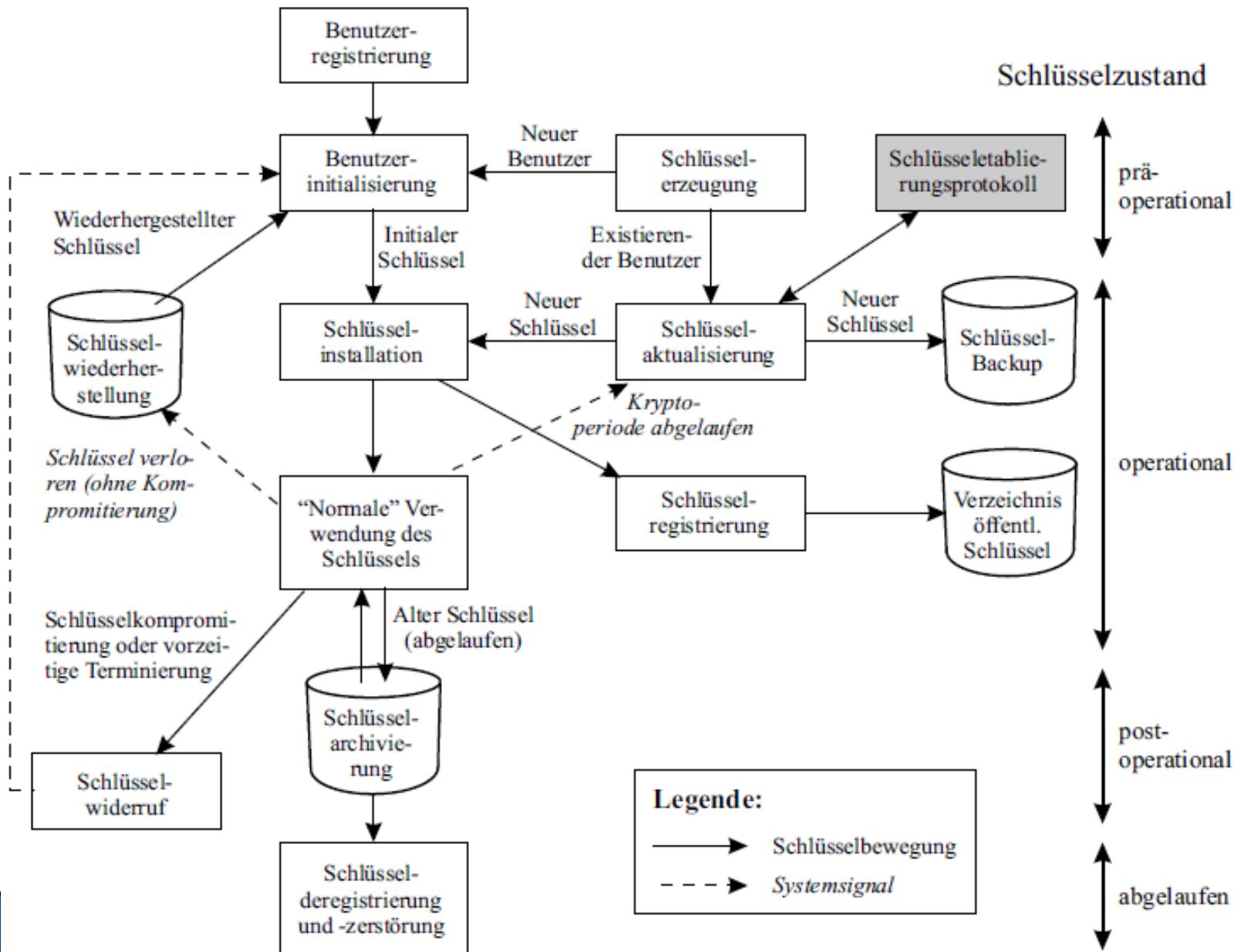
$K_T$	Master-Schlüssel des KTC
$K_{AT}$	gemeinsamer Schlüssel von $A$ und $T$
$K_{BT}$	gemeinsamer Schlüssel von $B$ und $T$
$E_{K_T}(K_{AT}, T)$	Zertifikat für $K_{AT}$
$E_{K_T}(K_{BT}, T)$	Zertifikat für $K_{BT}$

# TECHNIKEN UND PROTOKOLLE FÜR SCHLÜSSELMANAGEMENT

- Verteilung und Speicherung öffentlicher Schlüssel
  - Keine Vertraulichkeit jedoch Authentizität erforderlich
  - **Zertifizierung** öffentlicher Schlüssel erlaubt Überprüfung der Authentizität
  - Zertifikat eines öffentlichen Schlüssels ist eine Datenstruktur bestehend aus Daten- und Signaturteil
    - Datenteil enthält mindestens ein Identifikationsmerkmal (Name, Email-Adr. etc.) einer Partei und den öffentlichen Schlüssel dieser Partei
    - Signaturteil besteht aus der Signatur einer TTP über/unter den Datenteil, wodurch Zusammengehörigkeit von Identifikationsmerkmal und öffentlichem Schlüssel durch TTP bestätigt wird.
    - TTP hier in der Rolle als Zertifizierungsinstanz (Certification Authority – CA)
    - CA selbst benötigt ein asymm. Signierungsschlüsselpaar
    - Jeder Benutzer/Teilnehmer benötigt authentische Kopie des öffentlichen Signaturverifikationsschlüssels der CA um Signaturen prüfen zu können

159

# LEBENSZYKLUS DES SCHLÜSSELMANAGEMENTS



# INITIALE SCHLÜSSELBEZIEHUNGEN

- Zur Etablierung und Aktualisierung von Schlüsseln mit automatisierten Techniken benötigen Schlüsselmanagementsysteme einen **sicheren Kanal**.
- Deshalb setzen Schlüsselmanagementsysteme initiale Schlüsselbeziehungen zur Absicherung des Kanals voraus.
- Initialisierungsprozess umfasst nicht-kryptographische Prozeduren wie den persönlichen Transport des initialen Schlüsselmaterials

161

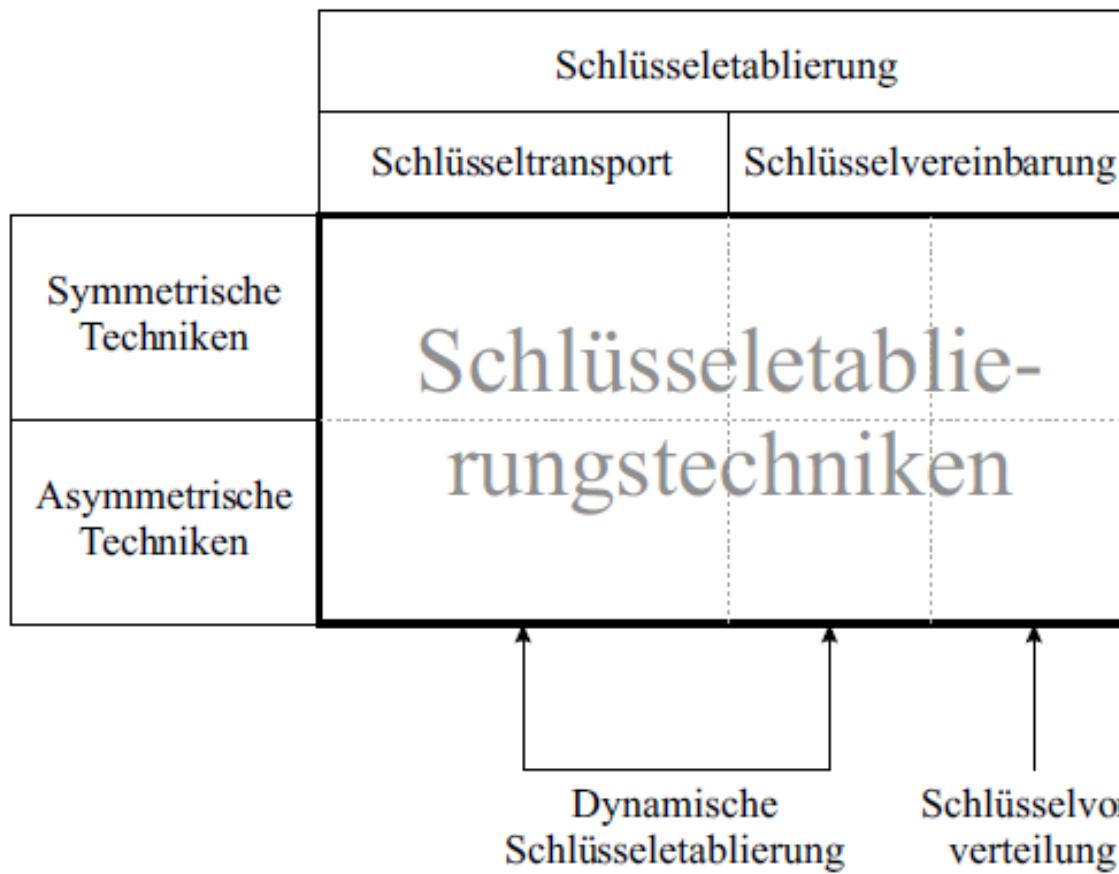
# SCHLÜSSELETABLIERUNGSPROTOKOLLE

- Schlüsseletablierung (Key Establishment):  
Verfügbarmachung eines geteilten Geheimnissen für zwei oder mehrere Parteien zur späteren kryptographischen Verwendung.  
Unterscheidung in
  - **Schlüsseltransport** (auch Schlüsselverteilung):  
Eine Partei erstellt Geheimnis und transferiert dieses sicher an Kommunikationsparteien
  - **Schlüsselvereinbarung**:  
Zwei (oder mehrere) Parteien leiten geteiltes Geheimnis als Funktion vorliegender Information ab.  
Jede Partei benötigt zur Ableitung des Schlüssels Information von allen anderen beteiligten Parteien.  
Idealerweise kann keine einzelne Partei den resultierenden geheimen Wert vorherbestimmen.

162

# SCHLÜSSELETABLIERUNGSPROTOKOLLE

- Typischerweise werden geheime **Sitzungsschlüssel** mit relativ kurzer Nutzungsdauer etabliert.
- ⇒ Keine Langzeitspeicherung von Schlüsseln für alle Kommunikationspartner erforderlich, sondern Etablierung von Sitzungsschlüsseln bei Bedarf.



# EIGENSCHAFTEN VON SCHLÜSSELETABLIERUNGSPROTOKOLLEN

## ■ Perfect Forward Secrecy

Ein Protokoll gewährleistet perfekte Vorwärtssicherheit (Perfect Forward Secrecy), wenn durch die Kompromittierung von Langzeitschlüsseln ehemalige Sitzungsschlüssel **nicht** kompromittiert werden.

## ■ Weitere Eigenschaften hier nicht näher betrachtet

164

# Schlüsseltransportprotokolle

- Es existieren unterschiedliche Protokolle sowohl auf symmetrischen als auch asymmetrischen Techniken basierend
- Symmetrische Nutzen KDC oder KTC
- Asymmetrischen nutzen i.d.R. digitale Signaturen
- Protokolle unterscheiden sich in spezifischen Eigenschaften Eigenschaften, die hier nicht näher betrachtet werden
- Recht bekannte Beispiele für Schlüsseltransportprotokolle mit symmetrischen Techniken:
  - Needham-Schroeder-Protokoll
  - Kerberos

165

# SCHLÜSSELVEREINBARUNGSPROTOKOLLE

- Im wesentlichen nur eine Basis-Protokoll bekannt, nämlich Diffie-Hellman
  - Es existieren verschiedene Varianten davon
- Verfahren war 1976 der erste jemals patentierte asymmetrische Algorithmus

166

# DIFFIE-HELLMAN-Schlüsselvereinbarung

## ■ Grundlagen

- Eine in der Kryptographie zum Einsatz kommende Einwegfunktion ist die modulare Potenzierung. Folgende Berechnung ist einfach:

$$a^x \bmod m$$

- Das zur modularen Potenzierung inverse Problem ist die Berechnung des diskreten Logarithmus einer Zahl. Dies ist ein schwere Problem:

Berechne  $a^x \bmod m = b \bmod m$  bei gegebenem  $a, b, m$ .

- Zum Beispiel folgt aus

$$3^x \bmod 17 = 15 \bmod 17$$

$$x = 6$$

- Die Lösung für Zahlen mit 1024 Bit Länge ist weitaus schwieriger.

# DIFFIE-HELLMAN-SCHLÜSSELVEREINBARUNG

- Die Beteiligten einigen sich vorab auf eine große Primzahl  $m$  und eine Zahl  $g$  (mit bestimmten Eigenschaften)
  - Diese Werte sind nicht geheim; können im Standard stehen
- Ablauf zwischen Parteien A und B
  - 1.) A wählt große zufällig Zahl  $x$ , die A geheim hält. Und sendet den folgenden Wert  $X$  an B

$$X = g^x \bmod m$$

- 2.) B wählt große zufällige Zahl  $y$ , die B geheim hält, und sendet den folgenden Wert  $Y$  an A

$$Y = g^y \bmod m$$

- 3.a) A berechnet
$$k = Y^x \bmod m = (g^y \bmod m)^x = g^{xy} \bmod m$$
- 3.b) B berechnet
$$k' = X^y \bmod m = (g^x \bmod m)^y = g^{yx} \bmod m$$
- $k=k'$  kann von A und B als gemeinsames Geheimnis genutzt werden.
- $X$  ist öffentlicher und  $x$  privater Schlüssel von A (analog  $Y$  und  $y$  für B)

# DIFFIE-HELLMAN-SCHLÜSSELVEREINBARUNG

## ■ Ablauf zwischen Parteien A und B

- 1.) A wählt große zufällig Zahl  $x$ , die A geheim hält. Und sendet den folgenden Wert  $X$  an B

$$X = g^x \bmod m$$

- 2.) B wählt große zufällige Zahl  $y$ , die B geheim hält, und sendet den folgenden Wert  $Y$  an A

$$Y = g^y \bmod m$$

- 3.a) A berechnet

$$k = Y^x \bmod m = (g^y \bmod m)^x = g^{xy} \bmod m$$

- 3.b) B berechnet

$$k' = X^y \bmod m = (g^x \bmod m)^y = g^{yx} \bmod m$$

## ■ Angreifer der mithört sieht die Wert $g, m, X, Y$

- Alleine damit kann er weder  $k$  noch  $k'$  berechnen
- Um  $k$  oder  $k'$  berechnen zu können, müsste der Angreifer den diskreten Logarithmus berechnen um  $x$  oder  $y$  zu erfahren

169

# FRAGEN?



170

# IT-SICHERHEIT

## 5. KRYPTOGRAPHIE

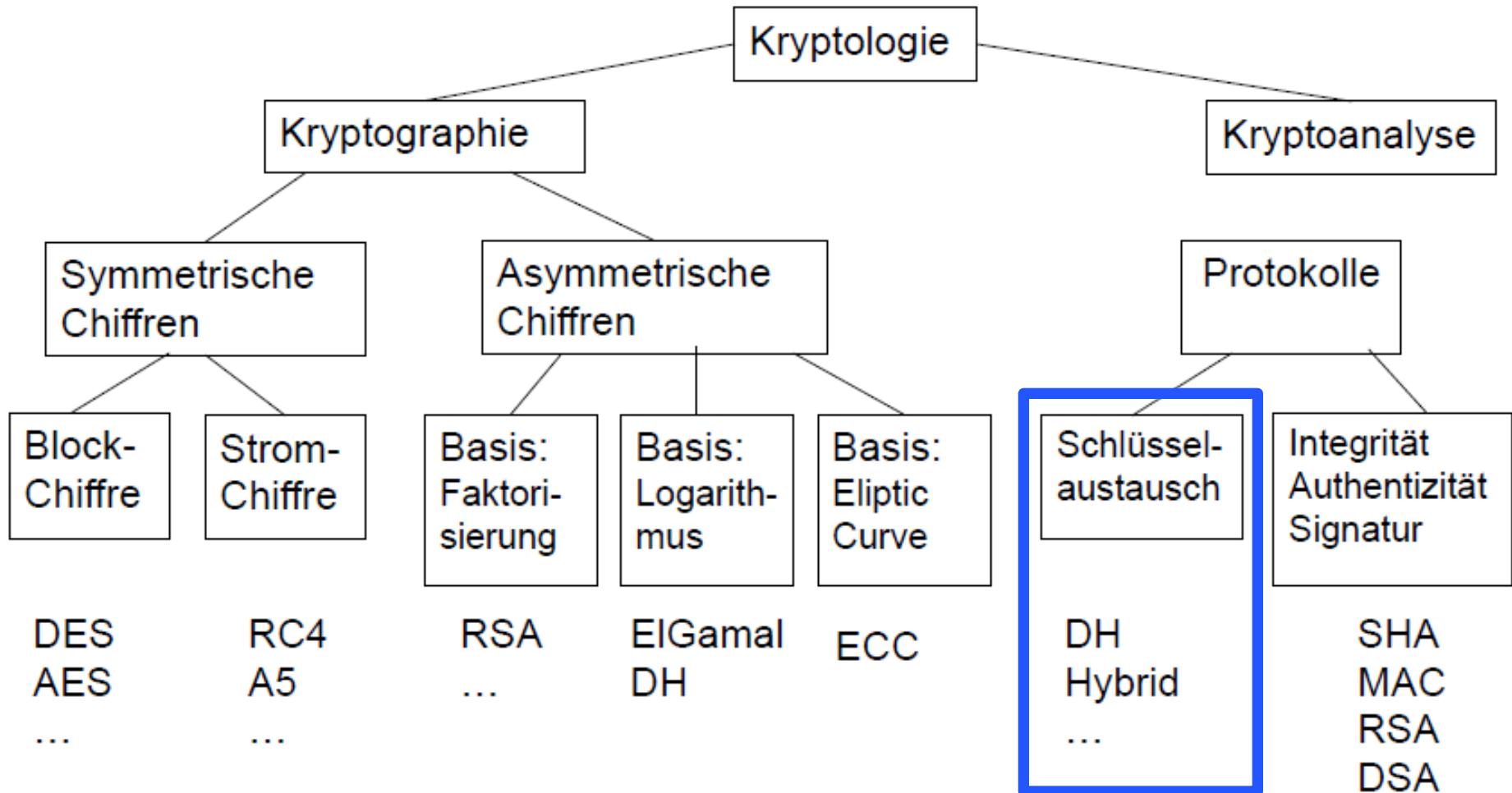
Wintersemester 2015/16

Prof. Dr. Michael Meier  
Informatik IV  
Arbeitsgruppe IT-Sicherheit



Universität Bonn  
E-Mail: mm@cs.uni-bonn.de

# DIAGRAMM



# SIGNIERTE NACHRICHT UND ZERTIFIKAT

-----BEGIN SIGNED MESSAGE-----

Hiermit bestelle ich folgende Waren:

10 Eier            Euro 2,00

1 Flasche Milch    Euro 1,50

1 Kasten Bier      Euro 15,00

Gesamtbetrag        Euro 18,50

Die Zahlung erfolgt bei Lieferung.

Michael Meier

-----BEGIN SIGNATURE-----

iQAAwUBOi9VLoBzbJXQK0fCYo1rMKCKrdhAn2Rs  
amogkkm+Off90L0W5RxUubfVuUFSXuv=

-----END SIGNED MESSAGE-----

-----BEGIN CERTIFICATE-----

Name: Michael Meier

Public key:

h833hd38dddajscbicme098k236egfkw74h544584h  
dbscldmrtpofjrkt0jedagaszw12geb3u4b=

Valid from: 19.11.2014

Valid until: 18.11.2017

Issuer: Einwohnermeldeamt Bonn

-----BEGIN SIGNATURE OF ISSUER-----

23j423vdsaz345kj435ekj4z2983734ijo23i72  
kj867wdbez2o074j51kdmcd1237t3rgbdbwdj=

-----END CERTIFICATE-----

# DREI ARTEN VON SIGNATUREN NACH SIGG

## ■ Signaturgesetz (SigG) vom 16.5.2001

- schafft rechtliche Rahmenbedingungen für den Beweiswert digitaler Signaturen

### **Elektronische Signatur**

Daten in elektronischer Form, die

- anderen elektronischen Daten beigefügt oder logisch mit ihnen verknüpft sind und die zur Authentifizierung dienen

### **Fortgeschrittene Signatur**

Daten in elektronischer Form, die

- ausschließlich dem Signaturschlüssel-Inhaber zugeordnet sind
- die Identifizierung des Signaturschlüssel-Inhabers ermöglichen
- mit Mitteln erzeugt werden, die der Signaturschlüssel-Inhaber unter seiner alleinigen Kontrolle halten kann
- mit den Daten, auf die sie sich beziehen, so verknüpft sind, dass eine nachträgliche Veränderung der Daten erkannt werden kann

### **Qualifizierte Signatur**

Daten in elektronischer Form, die

- die Anforderungen an eine fortgeschrittene Signatur erfüllen
- auf einem zum Zeitpunkt ihrer Erzeugung gültigen qualifizierten Zertifikat beruhen
- mit einer sicheren Signaturerstellungseinheit erzeugt werden

Sicherheit

4

# DREI ARTEN VON SIGNATUREN NACH SIGG

- Signaturgesetz (SigG) vom 16.5.2001
  - schafft rechtliche Rahmenbedingungen für den Beweiswert digitaler Signaturen

## Elektronische Signatur

Beispiel:

**E-Mail mit "Signatur"**

From: Michael Meier  
Subject: Beispiel

Das ist der Text.

--

Michael Meier  
Informatik 4  
Uni Bonn  
Uni Hamburg

### Fortgeschrittene Signatur

Beispiel:

**PGP-signierte E-Mail**

-----BEGIN PGP SIGNED MESSAGE-----

Hash: SHA1

Das ist der Text.

-----BEGIN PGP SIGNATURE-----

Version: PGP 8.0.2

iQA/  
AwUBP6wDdOFAIGFJ7x2EEQK9VgCg2Q4eQAz  
VIHP0HNFQ10eaXte96sAnR2p  
53T/SdevjXIuX6WOF5IXA44S  
=K3TO  
-----END PGP SIGNATURE-----

### Qualifizierte Signatur

Zertifikatausstellung nach  
Identitätsüberprüfung

sichere  
Signaturerstellungseinheit

Sicherheit

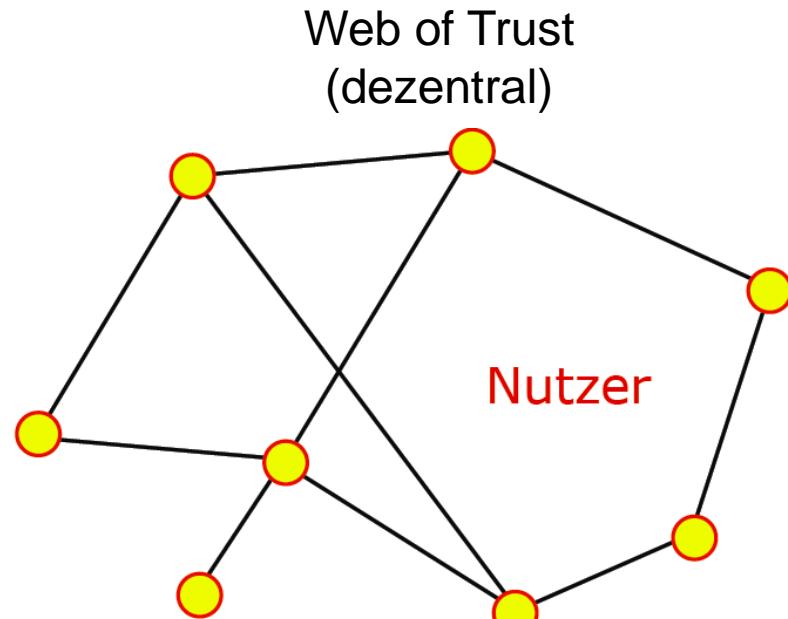


# ZWECK DER SCHLÜSSELZERTIFIZIERUNG

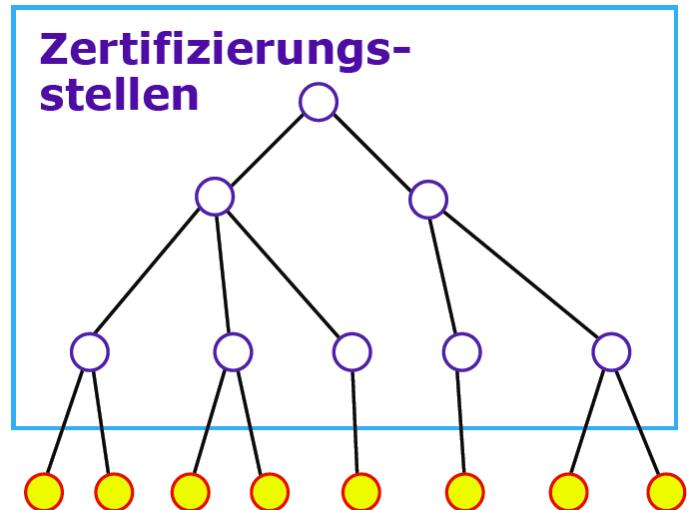
- Betrifft öffentliche Verifikationsschlüssel für digitale Signatur (und öffentliche Verschlüsselungsschlüssel)
- Zertifikat:
  - bestätigt die Zusammengehörigkeit von Verifikationsschlüssel und Benutzeridentität (bzw. Pseudonym)
  - Enthält selbst die Signatur des Zertifizierers
- Ohne Zertifikate:
  - Angreifer kann ein Schlüsselpaar generieren und einfach behaupten, dass dieser Schlüssel jmd. gehört.
  - Verifikationsschlüssel sind wertlos ohne Zertifikat (zumindest in einer offenen Welt)

# ZERTIFIZIERUNGSMODELLE

- Zur Verschlüsselung und Signierung wird asymmetrische Kryptographie verwendet
  - Zwei Schlüssel: Private und Public Key
  - Zwei Ansätze zur Zuordnung des Public Keys zu einer Person
    - Web of Trust (z.B. PGP, GnuPG)
    - Hierarchische Zertifizierung (z.B. SSL, S/MIME) auch Public Key Infrastruktur (PKI)

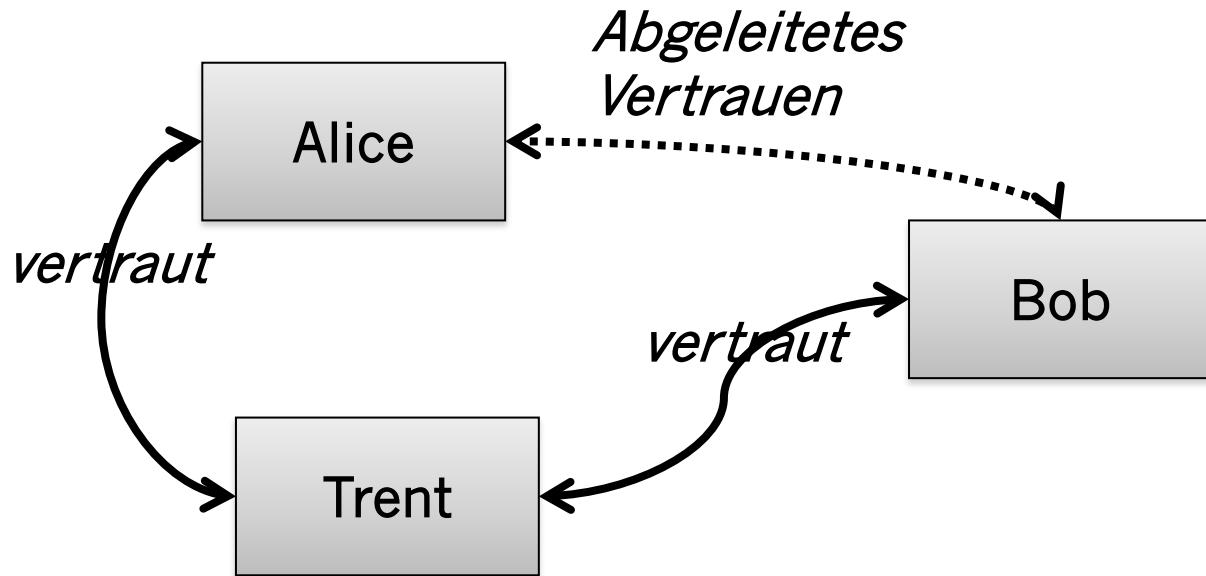


Hierarchische Zertifizierung  
(hierarchisch-zentral)

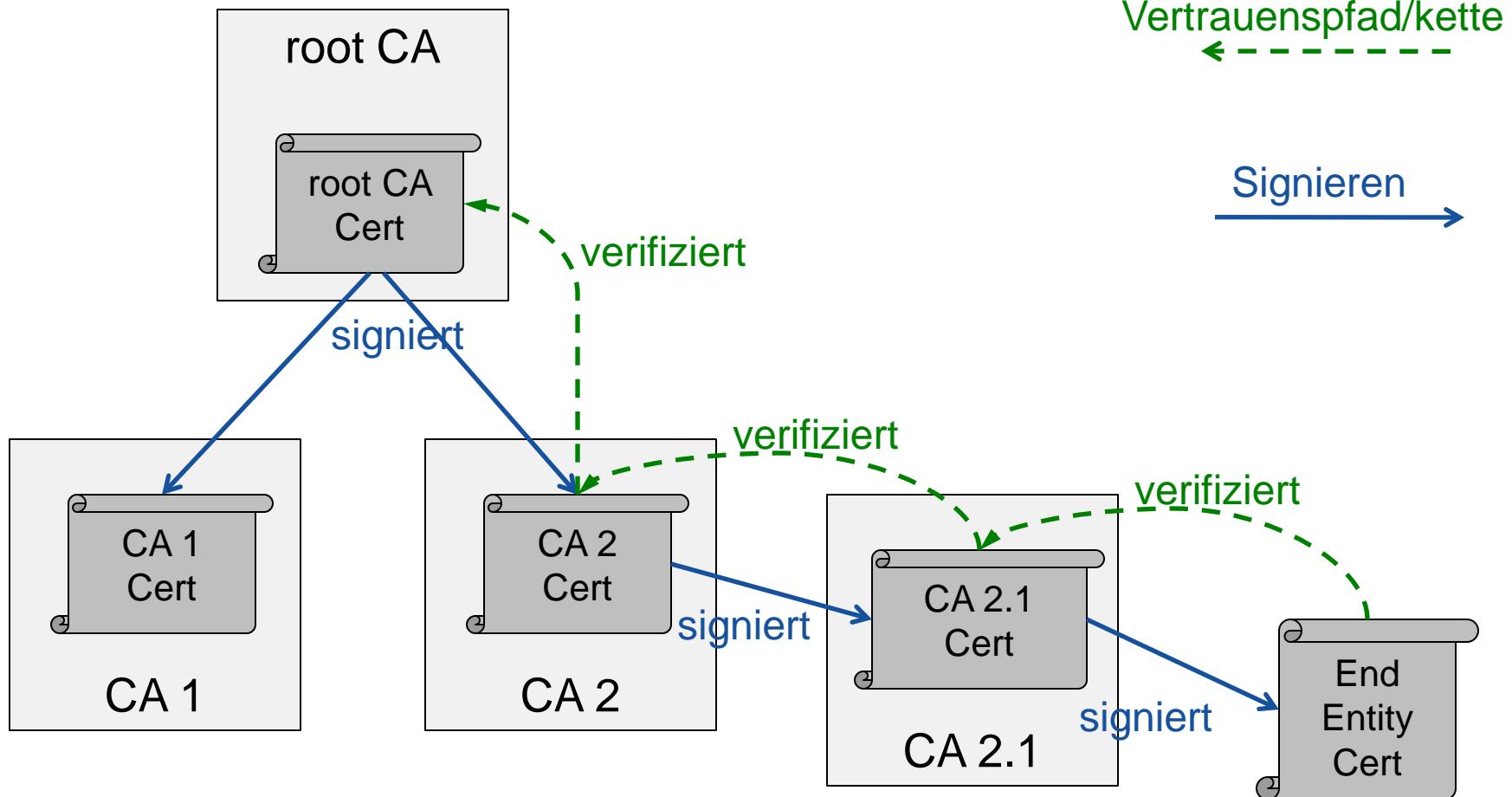


# VERTRAUEN IM WEB OF TRUST

- Ist Vertrauen transitiv?

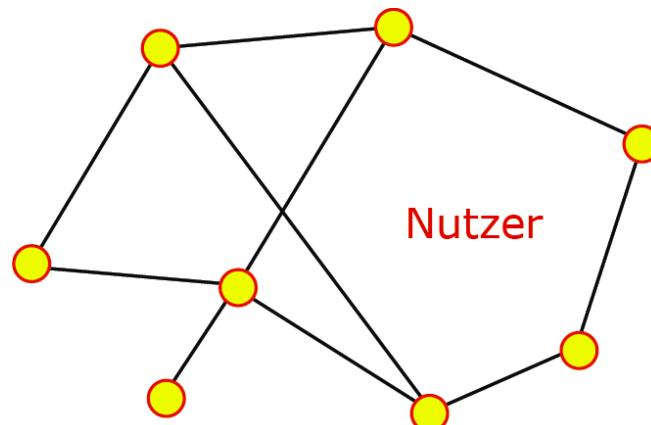


# VERTRAUENSPFADE IN HIERARCHISCHEN ZERTIFIZIERUNGSMODELLEN

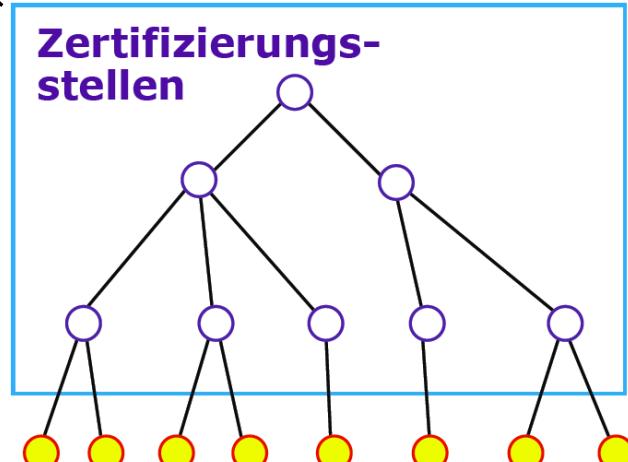


# ZERTIFIZIERUNGSMODELLE

## ■ Web of Trust (dezentral)



## ■ Hierarchische Zertifizierung (hierarchisch-zentral)



### ■ Vorteile:

- Einfache, flexible Nutzung
- Viele potentielle Zertifizierungsketten

### ■ Nachteile

- keine oder nur schwer erreichbare Beweisführung im Streitfall
- finden eines vertrauenswürdigen Pfades aufwendiger

### ■ Vorteile

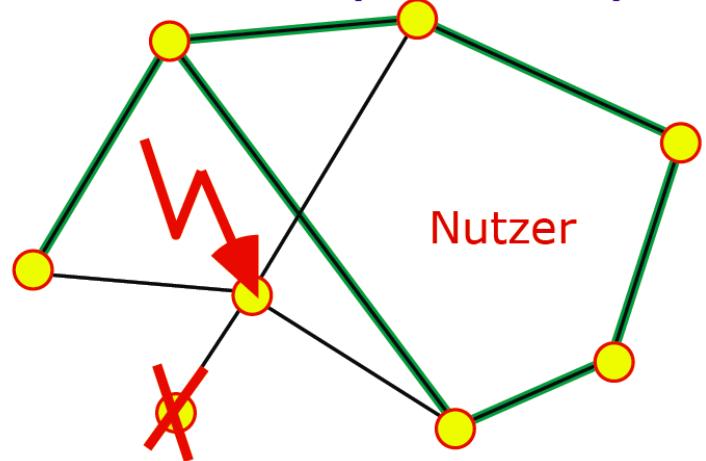
- klare Strukturen und Zurechenbarkeiten (wichtig im Streitfall)

### ■ Nachteile:

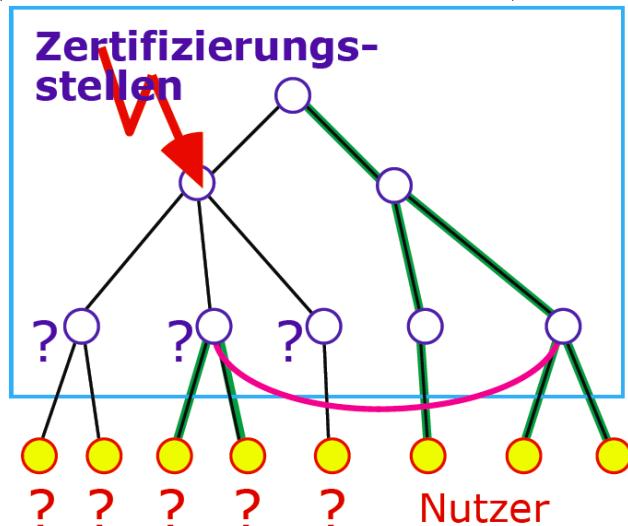
- Overhead durch Organisationsstruktur

# ZERTIFIZIERUNGSMODELLE

## ■ Web of Trust (dezentral)



## ■ Hierarchische Zertifizierung (hierarchisch-zentral)



### ■ Vorteile:

- Einfache, flexible Nutzung
- Viele potentielle Zertifizierungsketten

### ■ Nachteile

- keine oder nur schwer erreichbare Beweisführung im Streitfall
- finden eines vertrauenswürdigen Pfades aufwendiger

### ■ Vorteile

- klare Strukturen und Zurechenbarkeiten (wichtig im Streitfall)

### ■ Nachteile:

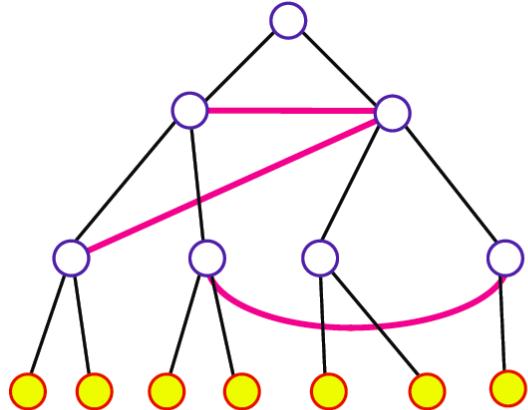
- Overhead durch Organisationsstruktur
- Anfällig gegen Fehlverhalten
- Cross Certification reduziert Fehlermöglichkeiten

# ZUSAMMENFASSUNG: ZERTIFIZIERUNGSMODELLE

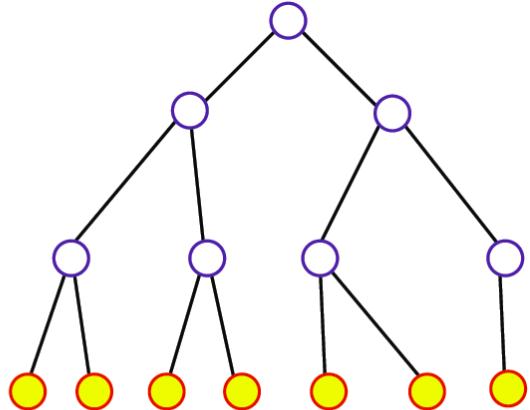
Haftung der  
Zertifizierers

ja

Vermaschter  
Graph



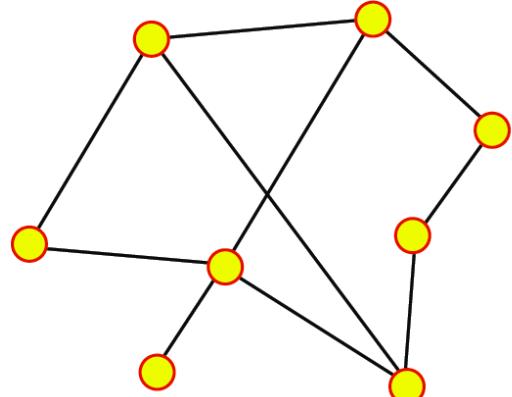
Baum  
(minimal zusammen-  
hängender Graph)



12

Reduzieren der  
Fehleranfälligkeit durch  
Cross Certification

Nein



# WAS GEHT MIT WEB OF TRUST?

- Kommunikation in geschlossener Gruppe
  - mit festgelegter Policy: sehr gut
- Offene Kommunikation
  - zunächst ohne rechtliche Relevanz: sehr gut
  - einfach nur vertraulicher und authentischer Nachrichtenaustausch zwischen den Kommunikationspartnern,
- eben Pretty Good Privacy
- Pretty Good Privacy  
<http://www.pgp.com/>
- Gnu Privacy Guard  
<http://www.gnupg.org/>



13

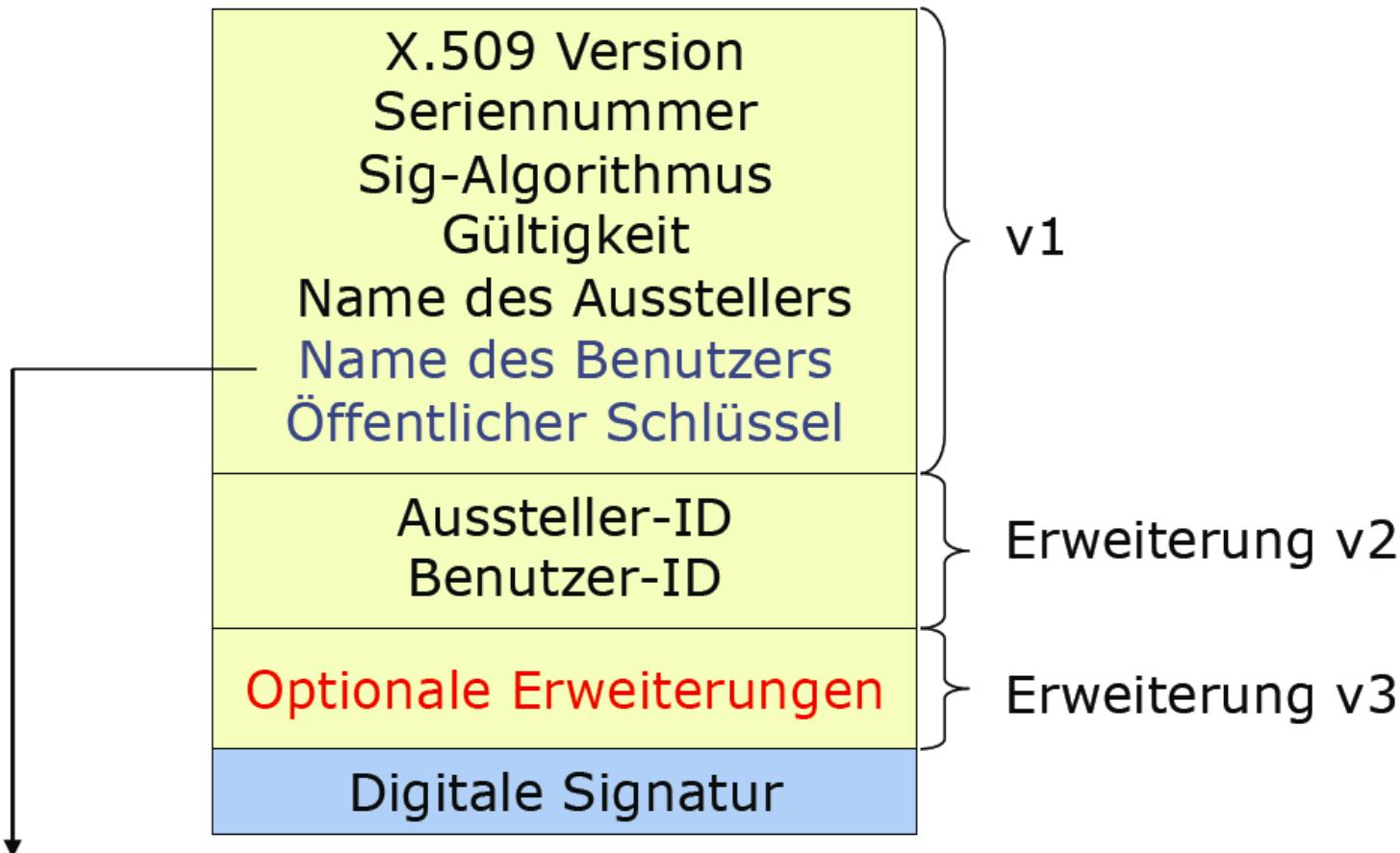
# X.509 ZERTIFIKATE

- Verbreiteter Standard für Zertifikate: X.509.v3 Format (RFC 2459)
- U.a. verwendet bei SSL und S/MIME
- S/MIME (Secure Multipurpose Internet Mail Extension)
  - Ursprünglich von RSA Data Security Inc.
  - S/MIME v3 im Juli 1999 als IETF-Standard verabschiedet
  - Internet Standards RFCs 2632-2634 (und weitere)
  - in die meisten E-Mail-Clients integriert
- SSL (Secure Sockets Layer)
  - auch: TLS (Transport Layer Security)
  - Verschlüsselung von TCP-Verbindungen
  - ursprünglich von Netscape für Browser entwickelt
  - heute in jedes moderne Betriebssystem integriert

14

## X.509 ZERTIFIKATE

- Festlegung eines standardisierten Formats für Zertifikate



Hierarchisch aufgebauter »distinguished name«, z.B.:

cn = Michael Meier, ou = Informatik, o = Uni Bonn, c = DE

# X.509 ZERTIFIKATE

- Erweiterungen in X.509v3
  - Art des Schlüssels, Anwendungsbereich
  - Alternative Namen für Inhaber und Aussteller
  - Einschränkungen bzgl. cross certification
  - Informationen bzgl. Sperrlisten (URL)
  - private ausstellerspezifische Erweiterungen
- Zertifizierungsprozess
  - Antrag bei der Registration Authority (RA)
  - Identitätsprüfung durch die RA
  - Zertifizierung durch die Certificate Authority (CA)
  - Ausgabe des Zertifikats an den Antragsteller
- Widerruf der Zertifikate
  - Certificate Revocation Lists (CRLs)
  - Online Certificate Status Protocol (OCSP)

16

# FAZIT ZU PUBLIC KEY INFRASTRUKTUREN

- Wissenschaftliche Fragen sind im Prinzip lange gelöst
- Dem PKI-Hype vor ca. 10 Jahren folgte Ernüchterung:
  - das Etablieren einer Unternehmens-PKI ist aufwändig
  - das Ausrollen von Zertifikaten und Signaturkarten ist aufwändig,
  - Nutzen war gering
  - Sperrlistenverwaltung, Trust-Center-Aufbau etc. schwierig
  - Unternehmensübergreifende PKI-Strukturen gibt es kaum

17

# SECURE SOCKET LAYER (SSL), RFC 6101

- 1995 als SSLv2 durch Netscape in Navigator 1.1 eingeführt
- 1996 Überarbeitung von SSLv2 zu SSLv3 (Netscape)
- ‚Quasi-Standard‘ für sichere TCP-Kommunikation
- 1999 Einführung von TLS (Transport Layer Security) v1.0
  - IETF Industrie-Standard, eng an SSLv3 angelehnt, aber inkompatibel zu SSLv3 (u.a. HMAC statt MAC, unterschiedliche Berechnung der Schlüssel)
- Open-Source-Implementation unter <http://www.openssl.org>
- RFC 5246 (TLS 1.2)
- RFC 2818 (HTTPS, SSL-basiertes HTTP)

# SSL/TLS KONZEPTE UND SCHUTZZIELE

- Basis: SSL/TLS setzt auf OSI-Schicht 4 (Transport) auf
- Client-Server-Kommunikationsparadigma

## Session:

- Aushandlung von zu verwendenden Krypto-Algorithmen und dezentrale Berechnung eines gemeinsamen Master-Secrets
- einer Session können mehrere Verbindungen zugeordnet sein
- alle Verbindungen der Session basieren auf dem MasterSecret

## Verbindung: entspricht einer TCP-Verbindung

- Individuelle Schlüssel pro Verbindung
- Verwendung der Algorithmen der zugeordneten Session
- Verbindungsaufbau: vorhandene Session oder neue nutzen

# SCHUTZZIELE VON SSL/TLS

## Sichere Client-Server-Kommunikation

- Authentifikation: verschiedene Varianten möglich:
  - wechselseitig, einseitig (meist nur Server), anonym
  - Mechanismen: X509-Zertifikate, MACs
- Integrität von Nachrichten
  - mit HMAC bzw. MAC-Verfahren: SHA-1
- Vertrauliche Datenübertragung
  - wählbare **Cipher-Suite**, (siehe nächste Folie)
  - symmetrische Verfahren
- Schlüsselaustausch:
  - Public-Key-Verfahren: RSA, DH\_DSS, DH-RSA, EC-DH

20

# SSL CIPHER SUITES

The screenshot shows two windows related to SSL/TLS cipher suite configuration:

- SSL: Edit Ciphers** window:
  - Tab bar: SSL2, **SSL3/TLS**, Extra SSL3/TLS
  - Section: Extra SSL3/TLS Cipher Suites
  - List of cipher suites with checkboxes and "Details..." buttons:
    - 256-bit AES encryption with RSA, DHE, and a SHA1 MAC
    - 256-bit AES encryption with DSA, DHE, and a SHA1 MAC
    - 256-bit AES encryption with RSA and a SHA1 MAC
    - 128-bit AES encryption with RSA, DHE, and a SHA1 MAC
    - 128-bit AES encryption with DSA, DHE, and a SHA1 MAC
    - 128-bit AES encryption with RSA and a SHA1 MAC
    - 168-bit Triple DES with RSA, DHE, and a SHA1 MAC
    - 168-bit Triple DES with DSA, DHE, and a SHA1 MAC
    - 56-bit DES encryption with RSA, DHE, and a SHA1 MAC
    - 56-bit DES encryption with DSA, DHE, and a SHA1 MAC
    - No encryption with RSA authentication and a SHA1 MAC
    - No encryption with RSA authentication and an MD5 MAC
  - Buttons: OK, Cancel- Cipher Details** window (opened from the "Details..." button for the first cipher suite):
  - Cipher: TLS\_DHE\_RSA\_WITH\_AES\_256\_CBC\_SHA
  - Encryption Algorithm: AES
  - Authentication Algorithm: RSA
  - Key Algorithm: DHE
  - Effective Key Size: 256
  - MAC Algorithm: SHA1
  - Other Attributes:
  - Buttons: OK, Cancel

**Cipher-Suite**  
beim SSL-Verbindungsaubau  
wird die zu  
verwendete Suite zwischen  
Client und Server  
im SSL-Handshake  
ausgetauscht

# SSL-PROTOKOLLE

Anwendungsprotokolle (HTTP, FTP, SMTP, SNMP)

Handshake-  
Protokoll

ChangeCipherSpec-  
Protokoll

Alert-  
Protokoll

ApplicationData-  
Protokoll

SSL-Record-Protokoll

TCP/UDP/IP

22

- **ChangeCipherSpec-Protokoll:** 1 Byte
  - Zeigt die Verwendung der vereinbarten Verfahren an
- **Alert-Protokoll:** dient zur Fehlerbehandlung, (2 Byte)
  - Erstes Byte: Schwere des Fehlers (1=Warnung; 2=Abbruch)
  - Zweites Byte: Fehlertyp (z.B. MAC-Prüfung schlägt fehl)

# SSL/TLS-HANDSHAKE-PROTOKOLL

23

## Aufgaben:

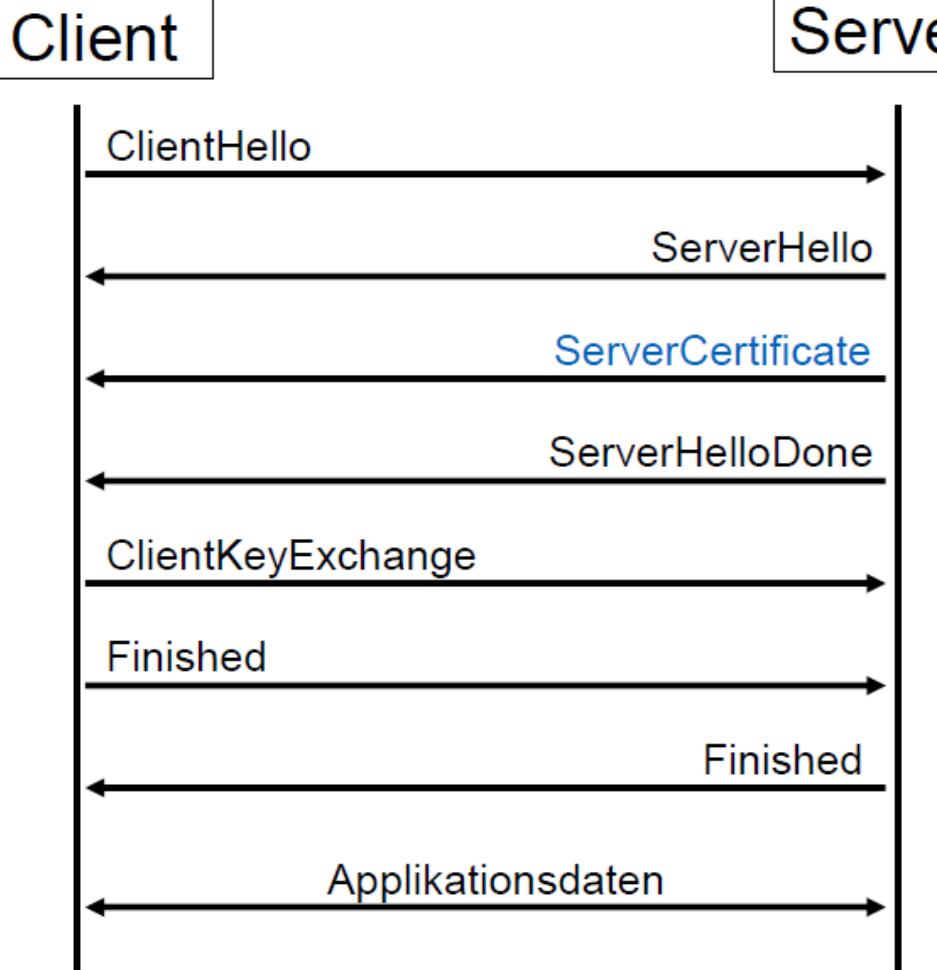
- Festlegen der zu verwendenden Krypto-Verfahren (Suite)
- Austausch von geheimer Basis-Information: Pre-Master Secret
- unidirektionale Verbindung: mit unterschiedlichen Schlüsseln
- Authentifikation mittels X509.v3-Zertifikaten
- Verwaltung von Sitzungsinformationen:
- Client/Server kann mehrere offene Sitzungen haben

## Verschiedene Modi:

- Server authentifiziert, Client anonym (u.a. bei Web-Portalen)
- Server und Client authentifiziert (über Zertifikate)
- Server und Client anonym

# SSL/TLS-HANDSHAKE-PROTOKOLL

Modus: Server authentifiziert,  
Client anonym



*ClientHello:*

- Protokollversion
- Randomzahl  $R_c$
- bekannte Verschlüsselungsmethoden
- bekannte Komprimiermethoden

*ServerHello:*

- Protokollversion
- Randomzahl  $R_s$
- gewählte Verschlüsselungsmethoden
- gewählte Komprimiermethoden

*ServerCertificate:*

- 24
- Liste der Zertifikate der Zertifizierungskette
  - Server-Public-Key, beim RSA-Verfahren

*ClientKeyExchange:*

- Pre-master Secret mit  
Server-Public-Key verschlüsselt

*Finished (Server bzw. Clientkennung):*

- Berechnen des Master-Secrets aus Pre-master und den Random-Werten  $R_c$ ,  $R_s$
- MD5-Hash-Wert mit Master-Secret über die bisher ausgetauschten Nachrichten
- SHA-Hash-Wert mit Master-Secret über die bisher ausgetauschten Nachrichten

# PROBLEME, GRENZEN VON SSL

- keine Verbindlichkeit: keine Signaturen
- häufig keine Forward Secrecy (mit DH ist das möglich)
- Vielzahl von vorkonfigurierten CAs: Vertrauen?
  - Werden Zertifikate geprüft? Warnungen ignoriert?
  - Stimmt Name im Zertifikat mit URL überein?
- Sichere Speicherung des Master-Secrets, der Session-Keys?
- Hacker-Angriffe auf CAs (u.a. DigiNotar 2011): Haftung für CA-Betreiber? (EU-Regelung in Vorbereitung)
- Häufig fehlende Zertifikatprüfung bei Nicht-Browser-Anwendungen (Cloud-Storage, PayPal Payment etc) durch fehlerhafte Nutzung von SSL-Libraries: anfällig gegen MitM Angriffe

25

# WELCHEN ZERTIFIZIERUNGSSTELLEN VERTRAUEN SIE?

- Firefox

Menu->Einstellung->Erweitert->Zertifikate->Zertifizierungsstellen

- Alle Builtin Object Token

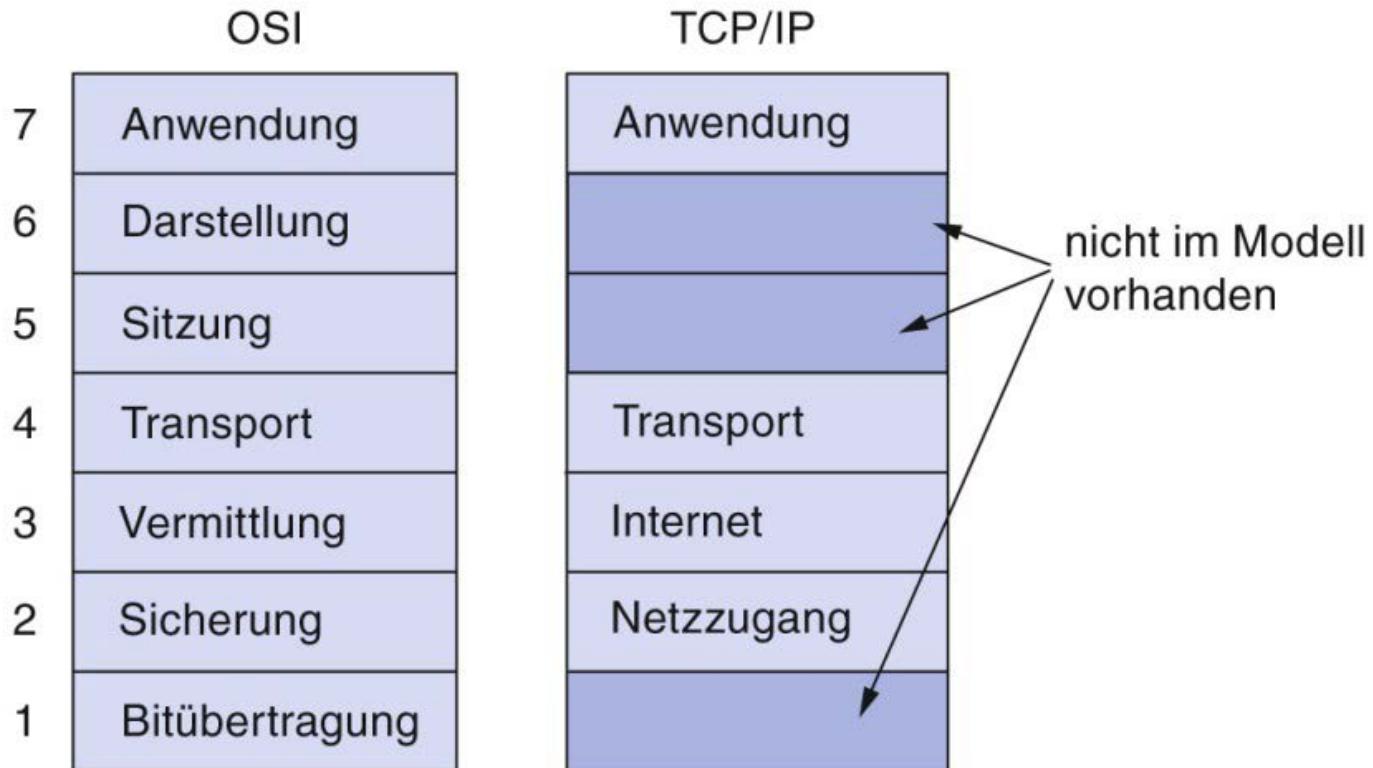
26

# KRYPTOGRAPHISCHE SICHERHEITSFUNKTIONEN NACH OSI-SCHICHTEN

Kommunikations-schicht im OSI-Referenzmodell	Sicherheitsfunktion
Anwendungsschicht	Pretty Good Privacy (PGP), S/MIME (Secure Multipurpose Internet Mail Extensions), Secure Shell (SSH)
Transportschicht	Secure Sockets Layer/Transport Layer Security (SSL/TLS)
Vermittlungsschicht	Authentication Header (AH) zur Integritätssicherung von Datagrammen, Encapsulated Security Payload (ESP) zur Verschlüsselung von Datagrammen
Schichten 1/2	Challenge Handshake Protocol (CHAP, Passwort), Encrypt Control Protocol (ECP), WiFi Protected Access (WPA) 2

27

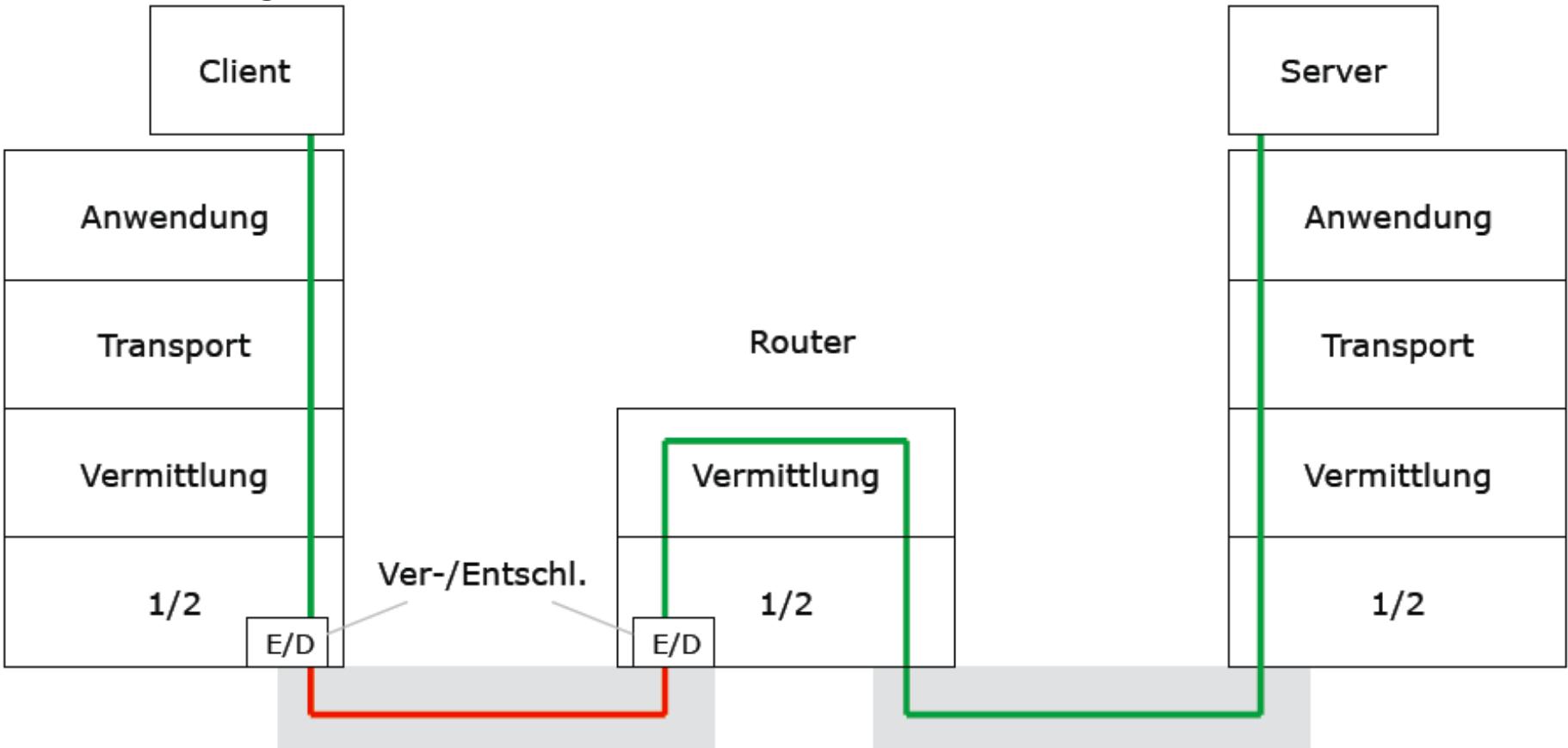
# WIEDERHOLUNG: OSI UND TCP/IP-REFERENZMODELL



# VERSCHLÜSSELUNG IN SCHICHT 1/2

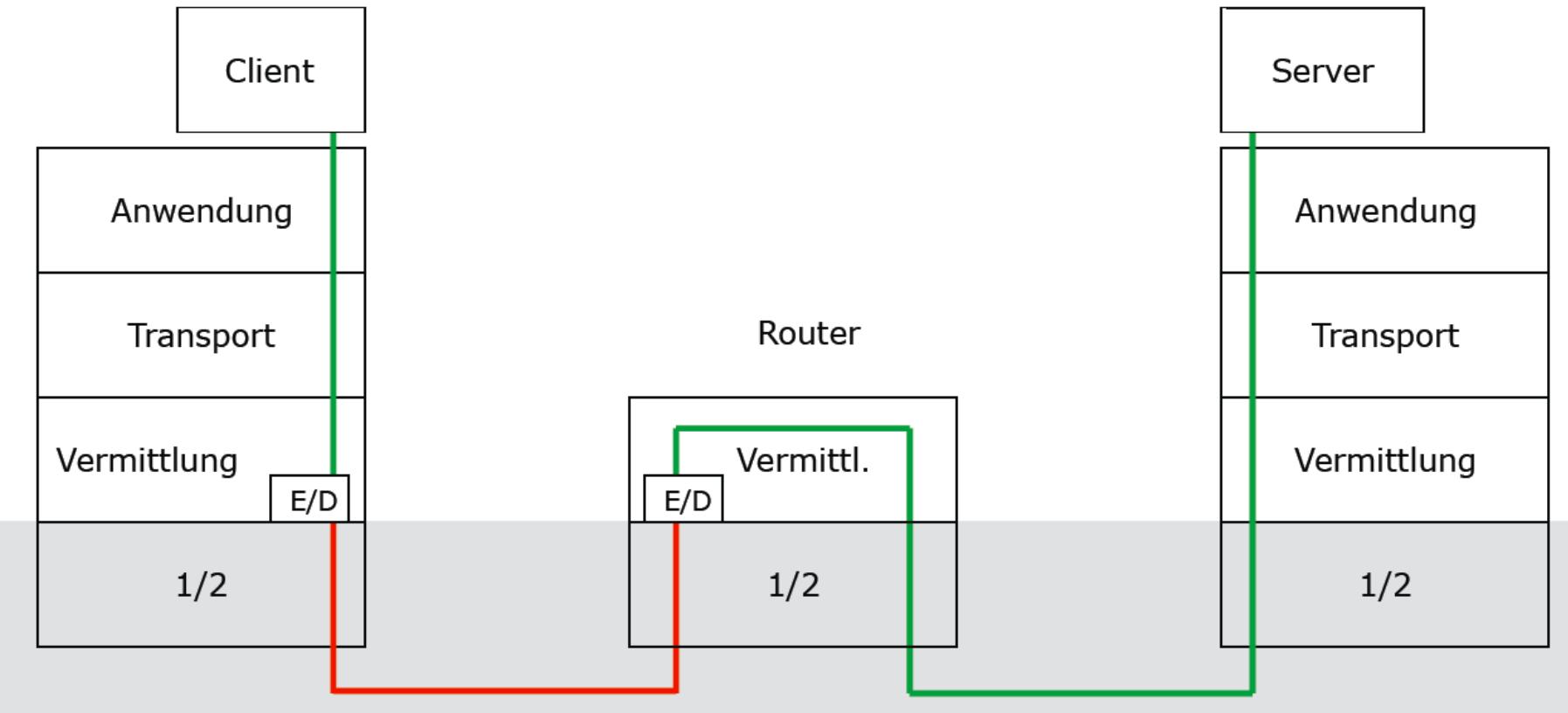
- Verschlüsselung nur bis zum nächsten Router (Verbindungsverschlüsselung)
  - Nicht alle Teilstrecken müssen verschlüsselt sein
  - Wenig Kontrolle durch den Endnutzer

29



# VERSCHLÜSSELUNG IN VERMITTLUNGSSCHICHT: IPSEC

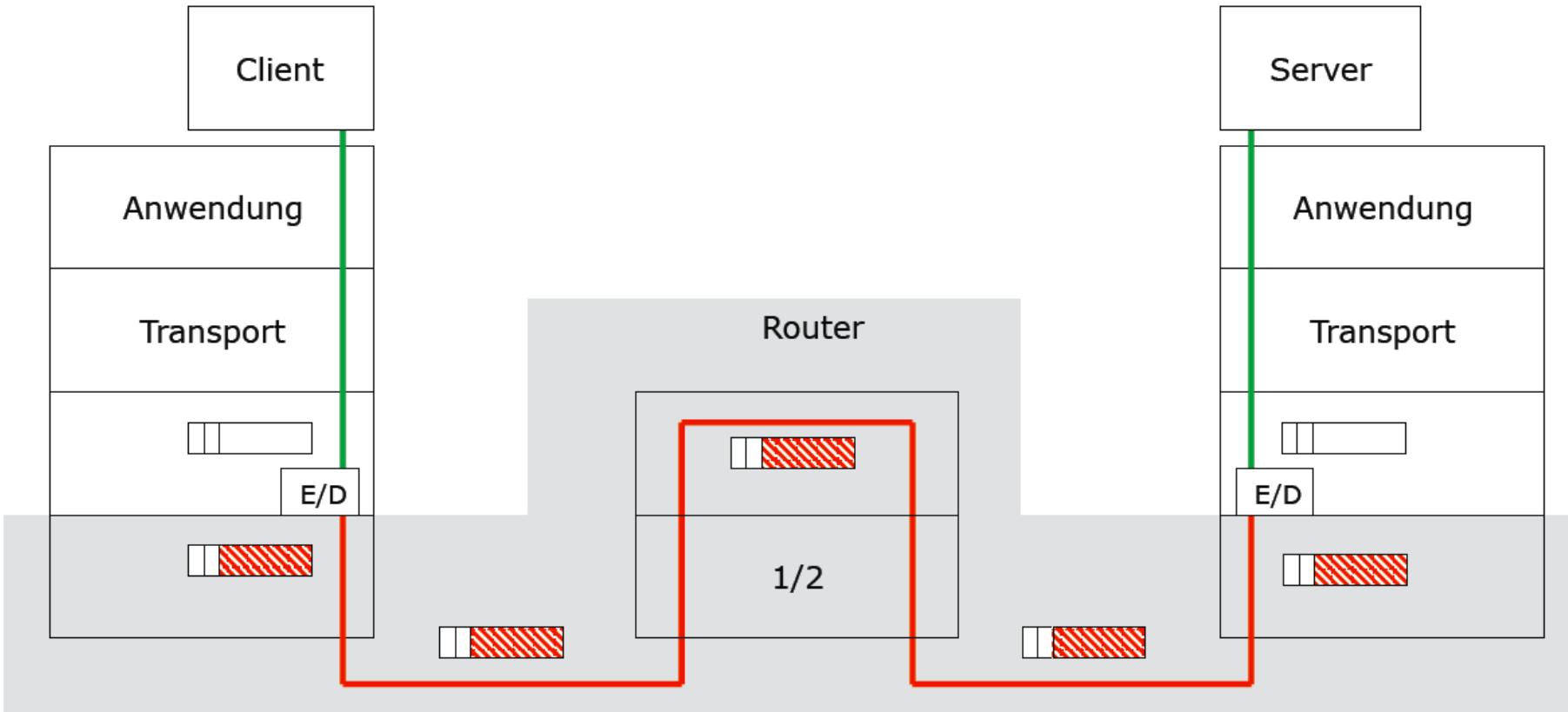
- Transportmodus
  - Verbindungs- und Ende-zu-Ende-Verschlüsselung möglich



30

# VERSCHLÜSSELUNG IN VERMITTLUNGSSCHICHT: IPSEC

- Transportmodus
  - Verbindungs- und Ende-zu-Ende-Verschlüsselung möglich

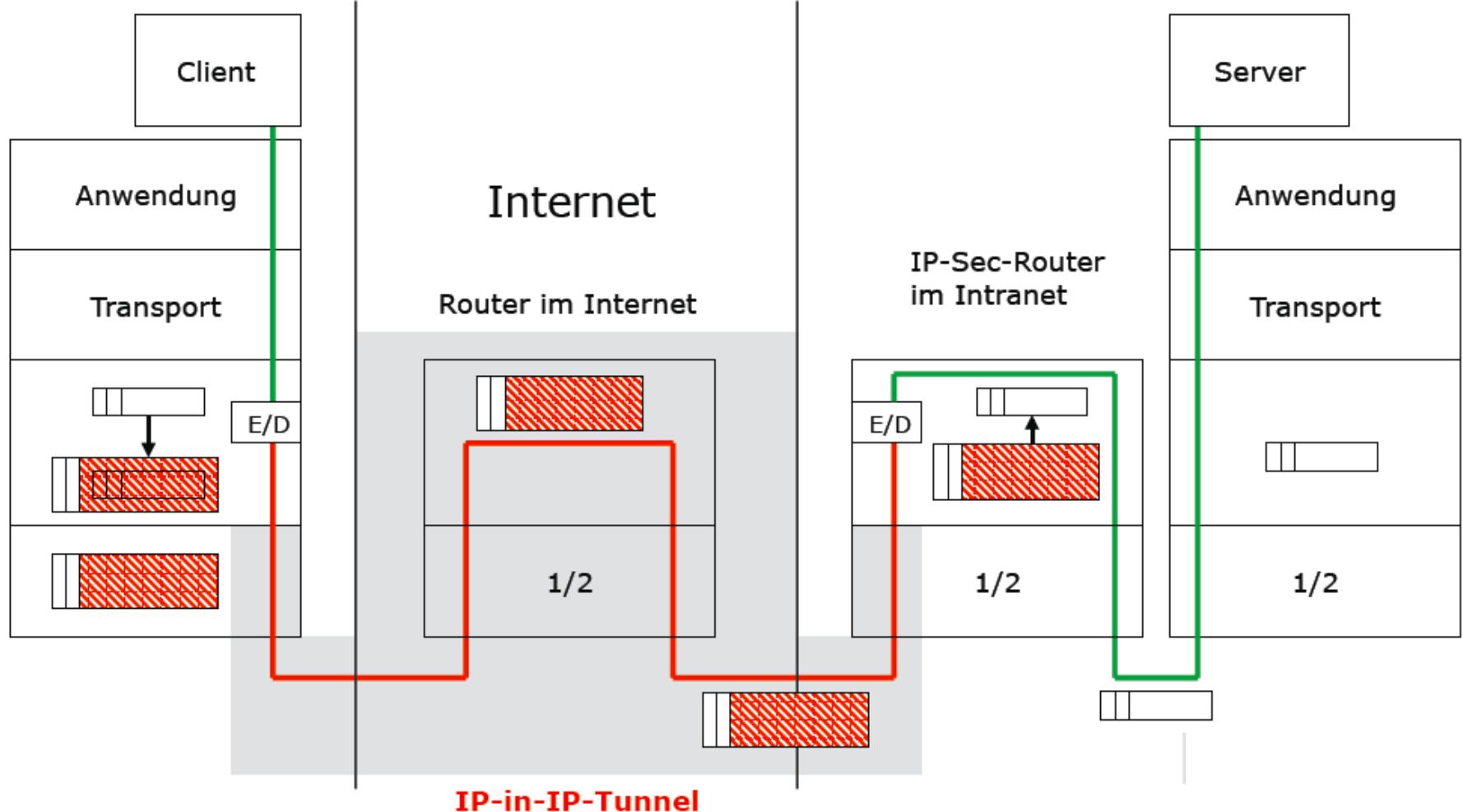


31

# VERSCHLÜSSELUNG IN VERMITTLUNGSSCHICHT: IPSEC

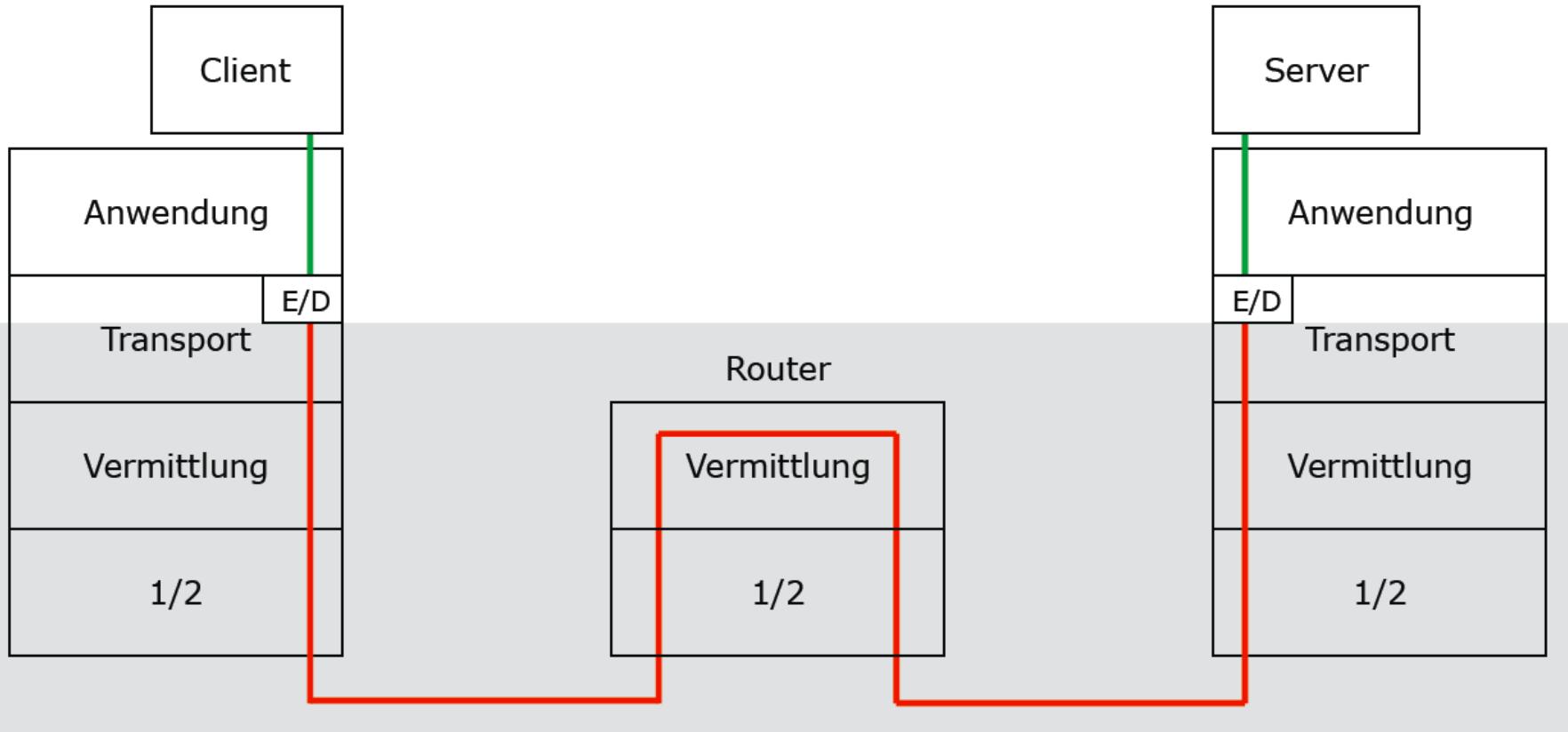
## ■ Tunnelmodus

- Momentane Hauptanwendung: Virtuelles Privates Netz (VPN)



# VERSCHLÜSSELUNG IN TRANSPORTSCHICHT: SSL/TLS

- Anwendung:
  - Verschlüsselung von TCP-Verbindungen



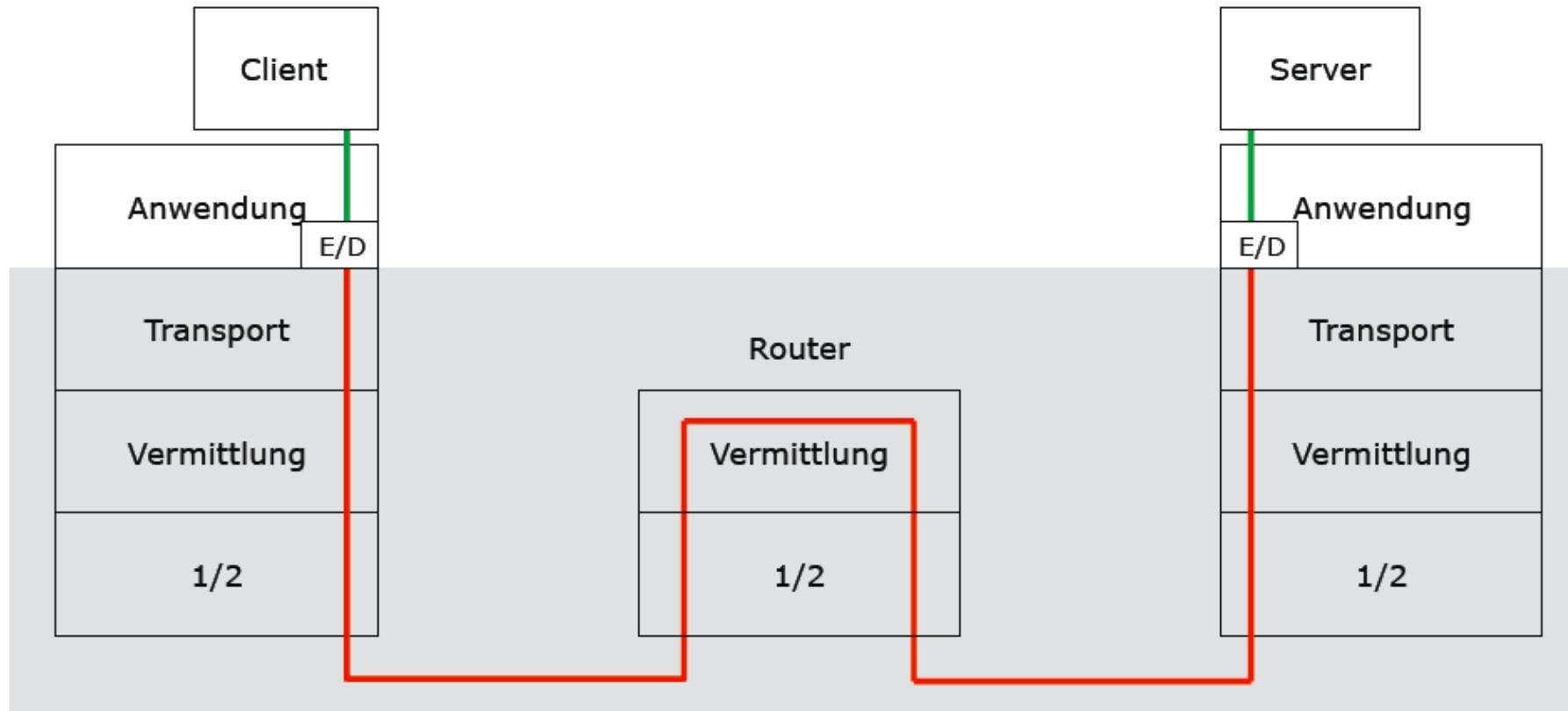
33

# VERGLEICH SSL - IPSEC

	<b>SSL</b>	<b>IPSec</b>
Komplexität	hoch	gering
Anwendungsnähe	hoch	gering
Für VPNs geeignet?	nein	ja
Für paketorientierte Dienste geeignet?	nein	ja
Für verbindungsorientierte Dienste geeignet?	ja	ja

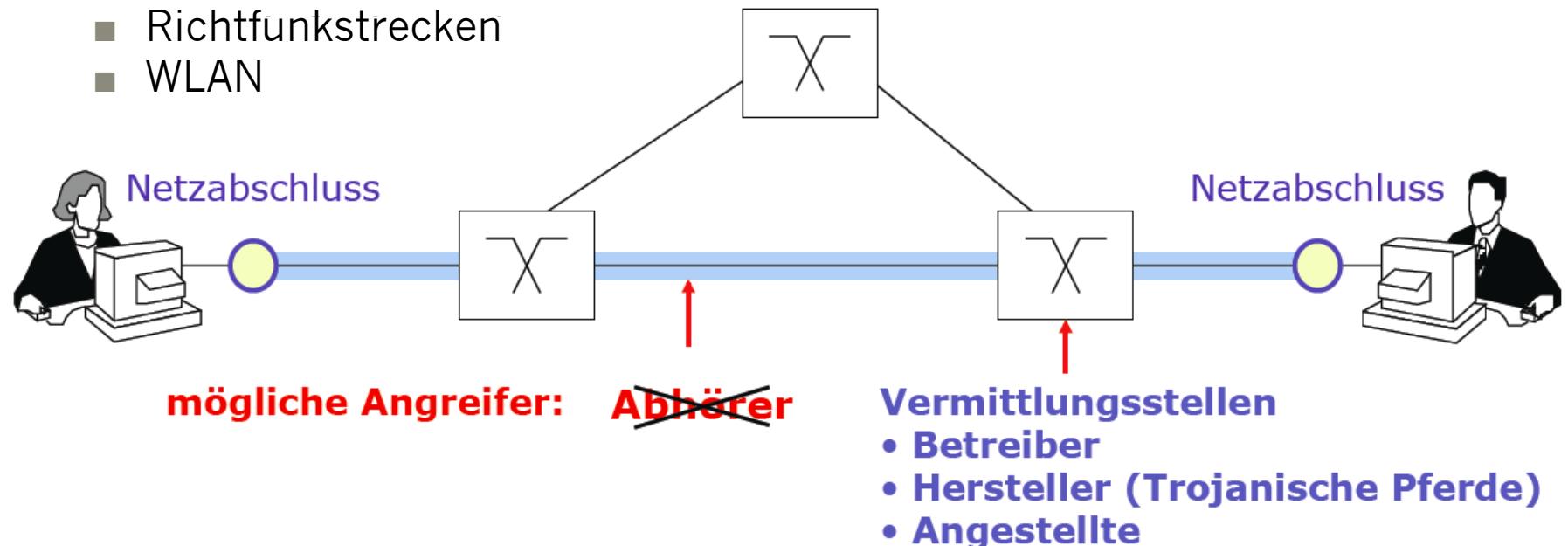
# VERSCHLÜSSELUNG IN ANWENDUNGSSCHICHT

- Ende-zu-Ende-Verschlüsselung zwischen Client und Server



# VERBINDUNGSVERSCHLÜSSELUNG

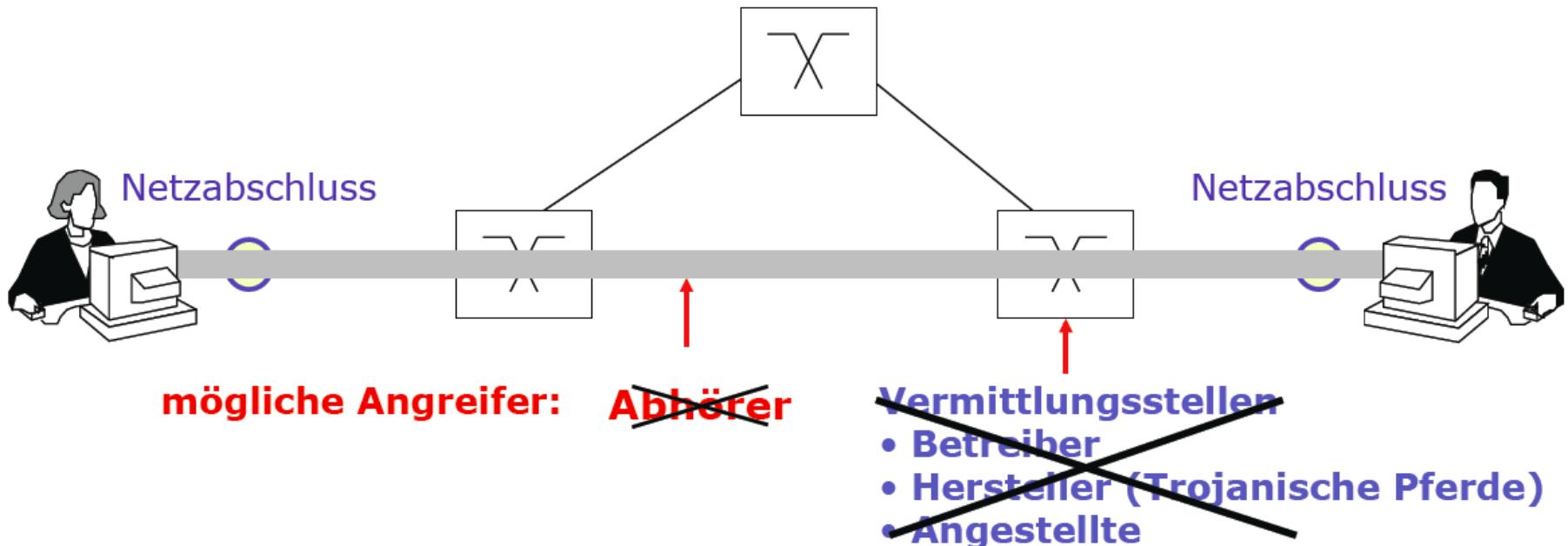
- Verbindungsverschlüsselung: (meist symmetrische Verschlüsselung)
  - zwischen Netzabschluss und Vermittlungsstelle
  - zwischen Vermittlungsstelle und Vermittlungsstelle
- In Vermittlungsstelle liegt Klartext vor
- Anwendungsbereiche:
  - Virtuelle Private Netze (VPN)
  - Leitungsverschlüsselung in Telekommunikationsnetzen
  - Richtfunkstrecken
  - WLAN



# ENDE-ZU-ENDE-VERSCHLÜSSELUNG

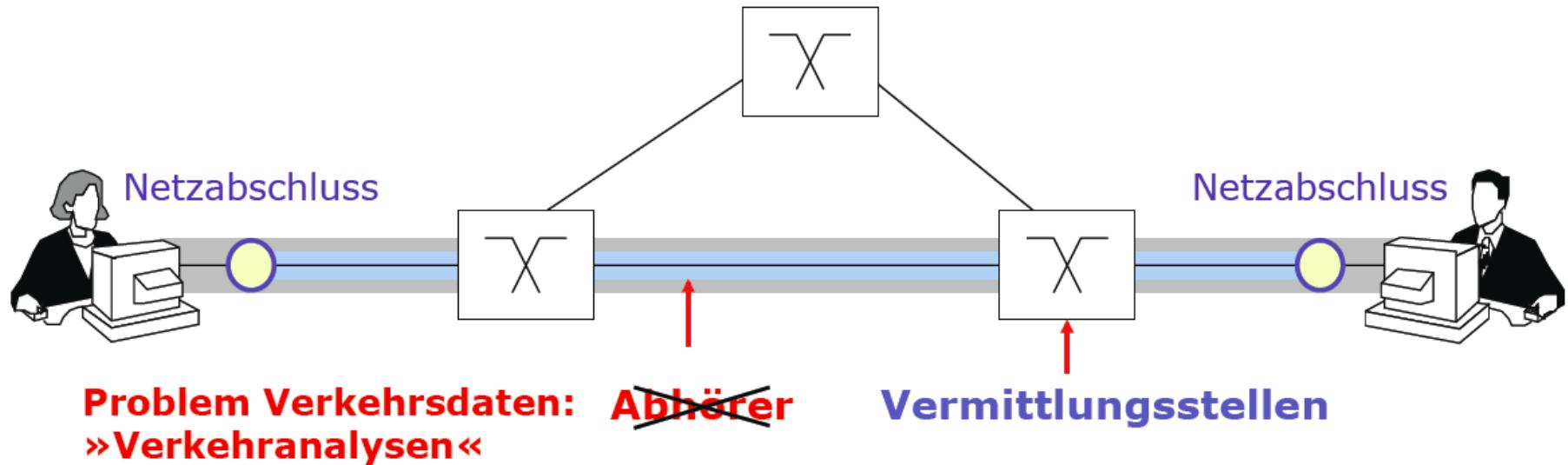
- Ende-zu-Ende-Verschlüsselung der Inhalte
  - von Endgerät zu Endgerät
- Anwendungsgebiete:
  - E-Mail-Verschlüsselung mit PGP oder S/MIME
  - Secure Sockets Layer (SSL)
- Adressierungsinformation kann nicht mit verschlüsselt werden

37



# VERBINDUNGS- UND ENDE-ZU-ENDE-VERSCHLÜSSELUNG

- Kombination von Verbindungs- und Ende-zu-Ende-Verschlüsselung
  - Ende-zu-Ende-Verschlüsselung allein schützt *nicht* die Adressierungsdaten vor Außenstehenden
  - zusätzliche Verbindungsverschlüsselung sinnvoll
- Restproblem Verkehrsdaten:
  - Netzbetreiber kann weiterhin feststellen, wer mit wem, wann, wie lange, wo, wie viel Information ausgetauscht hat



# DE-MAIL VS. GNUPG

- Siehe PDFs



39

# FRAGEN?

40

# IT-SICHERHEIT

## ZUSAMMENFASSUNG

Wintersemester 2015/16

Prof. Dr. Michael Meier  
Informatik IV  
Arbeitsgruppe IT-Sicherheit



Universität Bonn  
E-Mail: mm@cs.uni-bonn.de

# 1. INFORMATIONSTECHNIK UND NETZE

- CPU (x86) Betriebsmodi:
  - Real Mode
  - Protected Mode
    - Privilegienebenen / Ringe: User und Kernel Mode
    - Virtueller Speicher / Virtuelle Adressierung
- Festplatte, Partitionen, Sektoren, Master Boot Record (MBR)
  - MBR: Partitionstabelle, Master Boot Code
  - Bootsektor
- Interaktionssteuerung: Busy Waiting, Polling, Interrupt
- Ablauf beim Booten
- Wie greifen Programme auf Hardware zu? Systemcalls
- Wie erhält Betriebssystem die Ausführungskontrolle? Timer-Interrupt

# 1. INFORMATIONSTECHNIK UND NETZE

- Uni-, Multi-, Broadcast
- Protokollhierarchien/Netzwerkstacks: Protokoll, Schicht, Dienst, Schnittstelle
- Verbindungsorientierte, verbindungslose Kommunikation
- „Zuverlässige“ Kommunikation
- OSI-Modell, TCP-IP-Modell
- Broadcast-Netze, Medium Access Control, CSMA/CD
- IP, Fragmentierung, Adressen, Subnetze, Netzmaske, Präfix
- MAC/Ethernet- und IP-Adressen
- ARP, TCP (Socket, Port, Verbindungsaufbau), DNS
- Anwendungsprotokolle

3

## 2. GRUNDLEGENDES

- IT-System, Information, Daten, Objekte, Subjekte, Zugriffe
- Sicherheit und Verlässlichkeit
- Schutzziele
  - Vertraulichkeit, Integrität, Verfügbarkeit
- Schwachstelle, Verwundbarkeit, Exploit, Zero-Day-Exploit
- Bedrohung, Risiko, Angriff, Sicherheitspolitik
- Sicherheitsinteressen
- Security vs. Safety
- Mehrseitige Sicherheit, Inhaltsdaten, Verkehrsdaten
- Vertraulichkeiten, Verdecktheit, Anonymität, Unbeobachtbarkeit, Integrität, Zurechenbarkeit, Verfügbarkeit, Erreichbarkeit, Rechtsverbindlichkeit
- Prinzip: Informationelle Selbstbestimmung
- Schutzziele des Datenschutzes: Transparenz, Nichtverkettbarkeit, Intervenierbarkeit

4

## 2. GRUNDLEGENDES

- Schutzregel für personenbezogene Daten: Erlaubniserfordernis, Datensparsamkeit, Zweckbindung, Informationspflicht
- Common Criteria, Funktionalität und Vertrauenswürdigkeit/Qualität
- Wer kommt als Angreifer in Frage?
- Angreifermodell, Dimensionen bzw. Aspekte davon
- Notwendigkeit und Rolle von Vertrauen
- Annahmen, Beispiele
- Basisansätze für Sicherheitsmechanismen
  - Redundanz, Isolation, Ununterscheidbarkeit

5

### 3. PHYSISCHE SICHERHEIT

- Kann physische Sicherheit dauerhaft gewährleistet werden?
  - Verfügbarkeit, Integrität, Vertraulichkeit
- Seitenkanäle bei Chipkarten: Timing Attack, Power Analysis

## 4. AUTHENTIFIKATION

- Identifikation vs. Authentifikation
- Ansätze zur Authentifikation von Menschen: sein, haben, wissen
- Beispiele
- Authentifikation von Systemen

7

# 5. ZUGRIFFSKONTROLLE

- Zugriffsrecht
- Zugriffskontrollen
  - Benutzerbestimmbare Zugriffskontrolle (Discretionary Access Control)
    - Zugriffskontrollmatrix
    - Zugriffskontrolllisten (Access Control Lists)
    - Zugriffsausweise (Capabilities)
  - Systembestimmte Zugriffskontrolle (Mandatory Access Control)
    - Bell-La Padula Modell (Vertraulichkeits-Politik)
    - Biba Modell (Integritäts-Politik)
    - Chinese Wall Modell (Interessenskonflikte)
  - Jeweils Eigenschaften / Vor- und Nachteile
- Zugriffskontrolle in Unix/Linux
  - Rechte-Semantik für Verzeichnisse
  - Sticky-Bit, SetUserID, SetGroupID

## 5. ZUGRIFFSKONTROLLE (IM NETZ)

- Typen von Firewalls: Paketfilter, Circuit Level Gateway, Application Level Gateway (Proxies)
  - Einordnung OSI Schichten
  - Vor- und Nachteile
- Firewall-Architekturen
  - DMZ
- Grenzen von Firewalls

9

## 6. KRYPTOGRAPHIE

- Kryptographie, -analyse, -logie, Steganographie
- Informationstheoretisch sicher vs. Komplexitätstheoretisch sicher
- Kerckhoffs-Prinzip
- Angriffsarten:
  - Ciphertext-Only, Known-Plaintext, Chosen Plaintext
  - Brechen von Authentifikations- oder Verschlüsselungssystemen
- Symmetrische, asymmetrische, hybride
- Warum beschäftigt man sich mit Kryptoanalyse?
  - Klassisch (Mathem. oder Brute Force), Implementierung, Social Eng.
- Substitutionschiffre, Wie sicher? Wodurch brechbar?
- Transposition/Permutation
- Produktchiffre
- Strom- vs. Blockchiffre
- Funktionsweise Stromchiffre

10

# 6. KRYPTOGRAPHIE

- One-Time Pad / Vernam Chiffre
  - Funktionsweise, Eigenschaften (Uneingeschränkt / informationstheoretisch sicher; unpraktisch wegen Schlüssellänge)
- Blockchiffren Basisoperationen
- Konfusion: Beziehung zwischen Schlüssel und Chiffertext verschlüsseln
- Diffusion: ein Klartextzeichen beeinflusst möglichst viele Chiffretextzeichen (Avalanche/Lawinen-Effekt)
- DES
  - Grob „Innereien“, effekt. Schlüssellänge, Blockgröße, Sicherheit heute
- Triple DES
  - Effekt. Schlüssellänge, Meet-in-the-Middle-Angriff
- AES ( Rijndael)
  - Schlüssellänge(n), Blockgrößen

# 6. KRYPTOGRAPHIE

- Blockmodi
  - ECB, CBC, OFB, CFB, CTR
  - Jeweils Eigenschaften, Vor- und Nachteile
- Gedankenmodell zu asymmetrischen Chiffren
- RSA
  - Zugrundliegendes Problem
  - Verschlüsselung
  - Entschlüsselung
  - Blockgröße und Modul N
  - Raten von Klartexten / Wörterbuchangriff und Abwehr
- Hashfunktionen, vier Anforderungen
  - Geburtstagsparadoxon und Hashwert-Länge
- Message Authentication Code (MAC): Zweck, Ablauf
  - HMAC – Motivation und Ansatz
  - CBC-MAC

12

# 6. KRYPTOGRAPHIE

- Digitale Signatur
  - Mit RSA, Vorgehen und Ablauf
- Man in the Middle Angriff gegen asymmetrische Verfahren
- Schlüsselmanagement
  - Rolle/Aufgaben
  - Einfache Schlüssel-Managementmodell: Punkt-zu-Punkt-Komm., Schlüsselverteilzentren, Schlüsselübersetzungszentren, Vor- und Nachteile
  - Vertrauenswürdige dritte Parteien
  - Schutzbedarf von Schlüsseln
  - Techniken und Protokolle
    - Symmetrische Schlüsselzertifikate
    - Zertifikate öffentlicher Schlüssel

13

# 6. KRYPTOGRAPHIE

- Schlüsselmanagement (Forts.)
  - Schlüsseletablierungsprotokolle
    - Schlüsseltransport/verteilung vs. Schlüsselvereinbarung
    - Sitzungsschlüssel
    - Perfect Forward Secrecy
    - Diffie Hellman Schlüsselvereinbarung
  - Signatur vs. Zertifikat
  - Schlüsselzertifizierung, Zweck
  - Zertifizierungsmodelle: Web of Trust, Hierarch. Zertifizierung Public Key Infrastruktur; Beispiel-Anwendungen davon, Vor- und Nachteile
  - Überprüfung Zertifikatskette, Root-Zertifikat
  - X.509 Zertifikate
- SSL, Rootzertifikate
- Kryptofunktionen nach OSI-Schichten
  - ...

14

# FRAGEN?

15

# AUSBLICK

- Bachelor:
  - Sommersemester: VL Reaktive Sicherheit (2+2 SWS)
  - Jedes Semester: PG
- Master
  - Sommersemester: VL IT Security (2+2 SWS)
  - Jedes Semester: Lab, Seminar
- Abschlussarbeiten (Bachelor, Master)
  - Vorgehen: finde Interessengebiete und Themenangebote der Mitarbeiter der AG ITSEC/Meier auf deren Webseiten
- Beste Klausur:
  - Freiticket zur Fachtagung Sicherheit 2016, 4.-7.4.2016, Uni-Club Bonn  
<https://sicherheit2016.de>

16

# FRAGEN?

17