Rechnernetze und verteilte Systeme (BSRvS II)

Prof. Dr. Heiko Krumm

FB Informatik, LS IV, AG RvS
Universität Dortmund

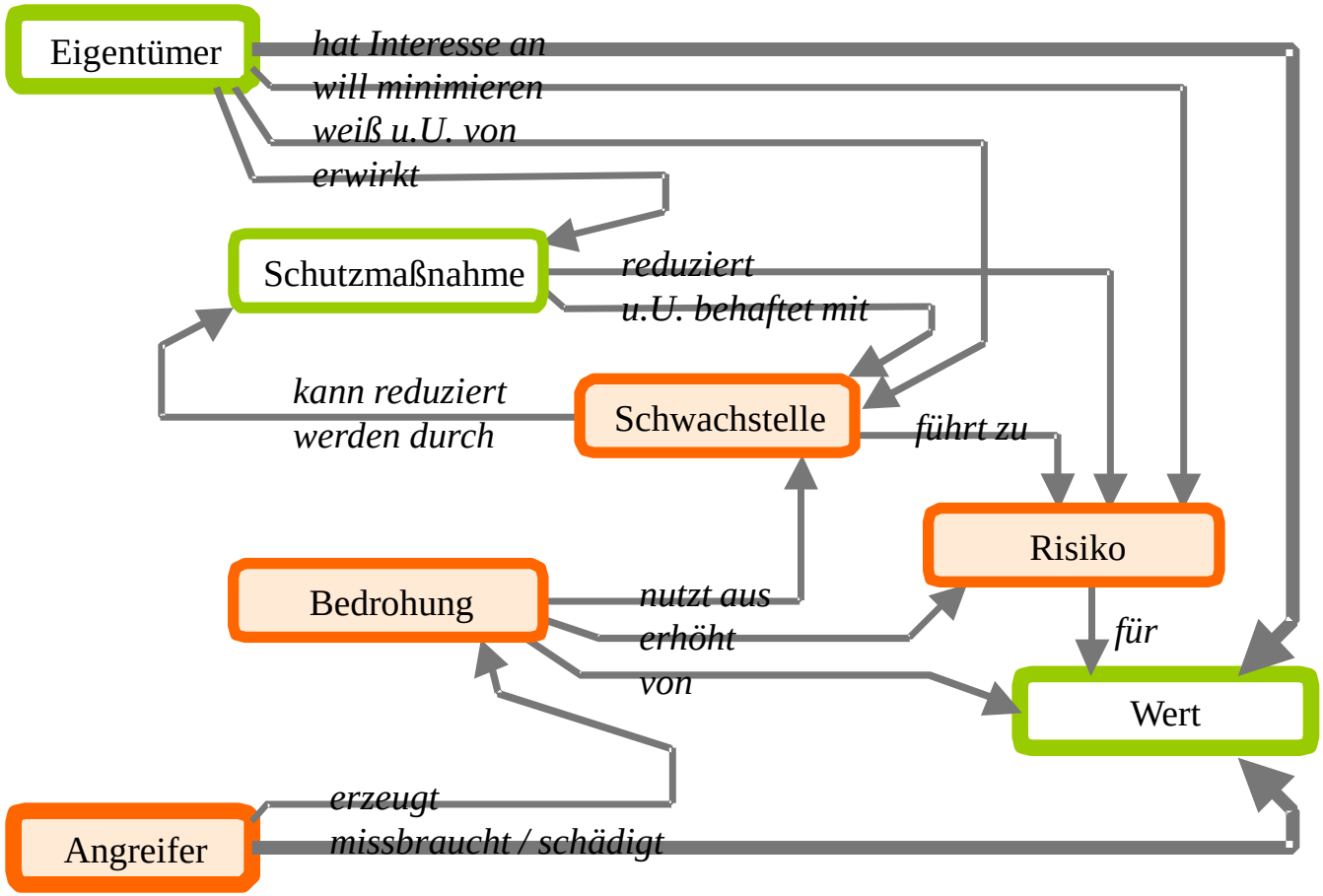
- Sicherheitsziele
- Kryptographie abstrakt
- Authentifikation
- Integrität
- Schlüsselverteilung und Zertifikate
- Firewalls
- Angriffe und Gegenmaßnahmen
- IPsec

- **Computernetze und das Internet**
- **Anwendung**
- **Transport**
- **Vermittlung**
- **Verbindung**
- **Multimedia**
- **Sicherheit**
- **Netzmanagement**
- **Middleware**
- **Verteilte Algorithmen**

Kap. 7: Sicherheit im Netz

Lernziele:

- Prinzipien der Sicherheit im Netz
  - Kryptographie und Nutzungen, die über Vertraulichkeitsschutz hinausgehen
  - Authentifikation
  - Nachrichtenintegrität
  - Schlüsselverteilung
- Sicherheit in der Praxis
  - Firewalls
  - Sicherheitsfunktionen in den Kommunikationsschichten



Kap. 7: Übersicht

7.1 Sicherheitsziele

7.2 Kryptographie abstrakt

7.3 Authentifikation

7.4 Integrität

7.5 Schlüsselverteilung und Zertifikate

7.6 Firewalls

7.7 Angriffe und Gegenmaßnahmen

7.8 Sicherheit in den verschiedenen Kommunikationsschichten

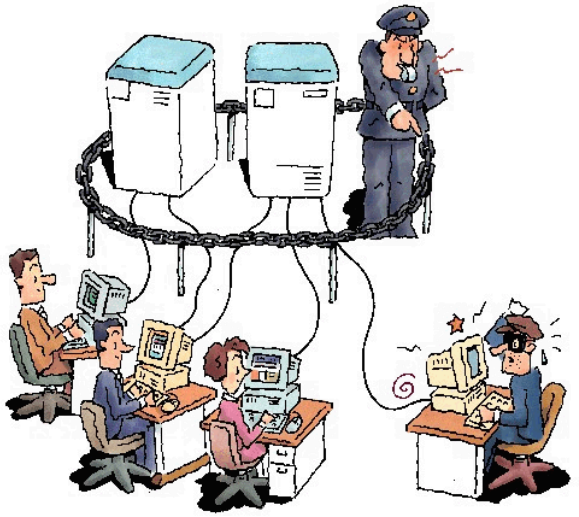
Sicherheitsziele

Vertraulichkeit *Die drei immer genannten Hauptziele*
Integrität
Verfügbarkeit

Anonymität *Es gibt weitere Ziele. Ziele können gegensätzlich sein*
Nachvollziehbarkeit / Zurechenbarkeit
…

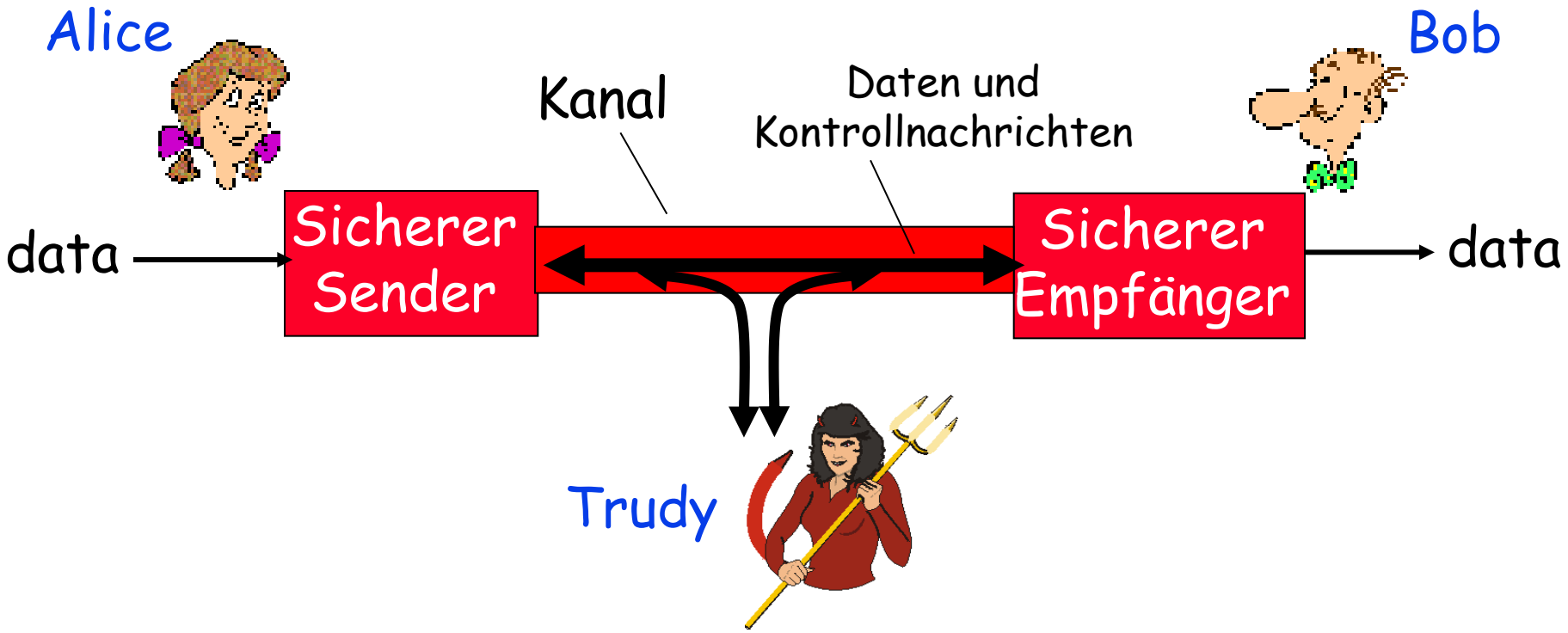
Authentifikation *Die beiden grundlegenden Hilfsdienste*
Autorisierung

Im Netz:
Nachrichtenvertraulichkeit / Integrität
Nachrichten--Absenderauthentifikation,
Empfängerauthentifikation



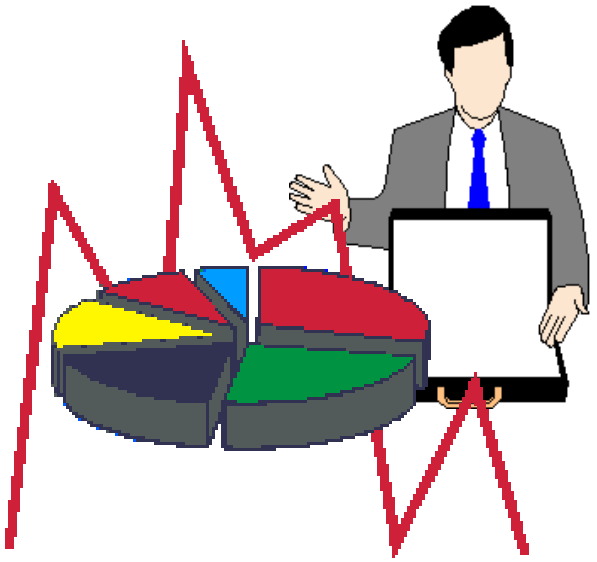
Freunde und Feinde: Alice, Bob, Trudy

- In der Welt der Netzsicherheit wohlbekannt
- Bob und Alice (befreundet!) wollen sicher kommunizieren
- Trudy (der Eindringling) kann Nachrichten abfangen, löschen, verändern, einschleusen



Wer kann Bob und Alice sein?

- … natürlich *real-life* Bobs und Alices!
- Web-Browser und Server, die elektronische Transaktionen asusführen (e.g., On-line-Shop Einkauf)
- On-line Banking-Client und Server
- DNS-Server
- Router, die Routingtabellen aktualisieren
- weitere Beispiele?



Es gibt aber überall auch bad Guys (und Girls)!

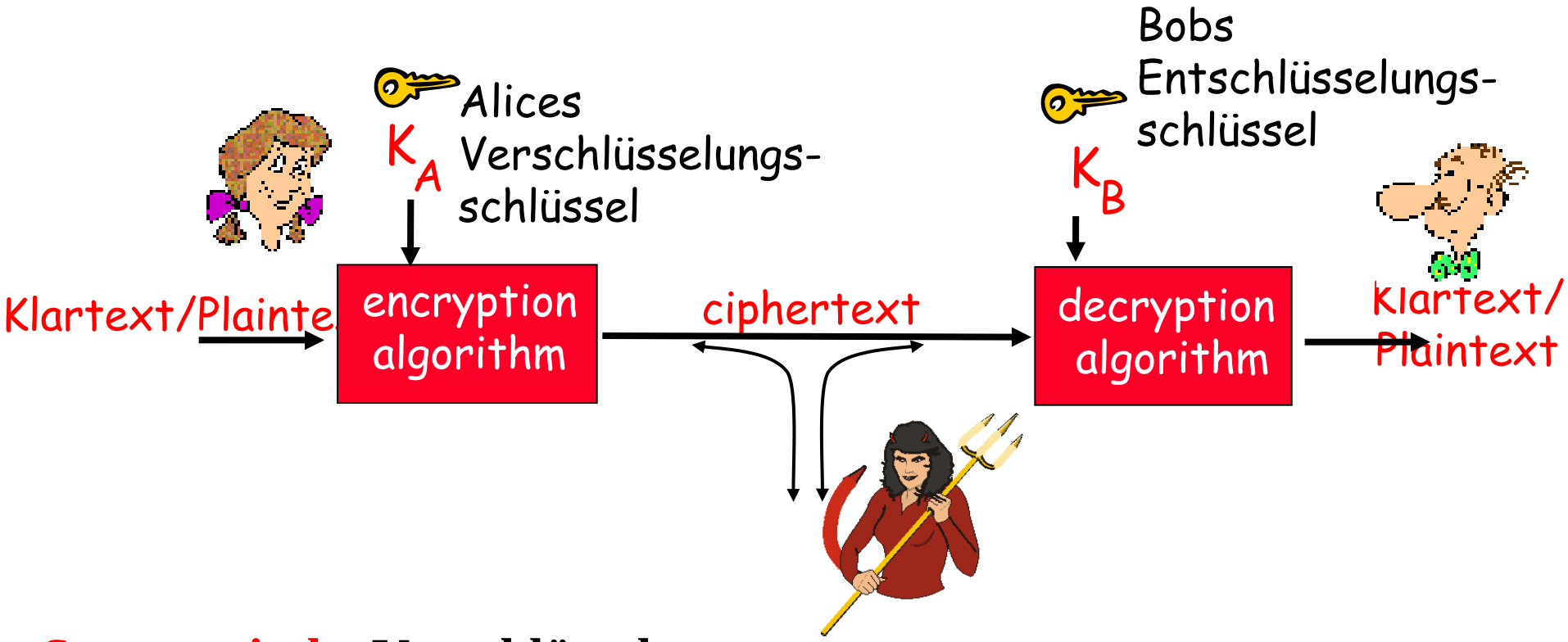
F: Was kann ein “bad Guy” tun?

A: Jede Menge!

- *Abhören*
- aktiv neue Nachirchten einfügen / unterschieben
- *Maskerade:* fälschen (spoof) der Quelladresse eines Pakets (oder anderer Kontrollfelder)
- *Sitzungsübernahme (Hijacking) / Verbindungsübernahme*
- *Verfügbarkeitsattacke (Denial of Service / DoS-Attacke)*

darüber später mehr……

Kryptographie abstrakt



Symmetrische Verschlüsselung:

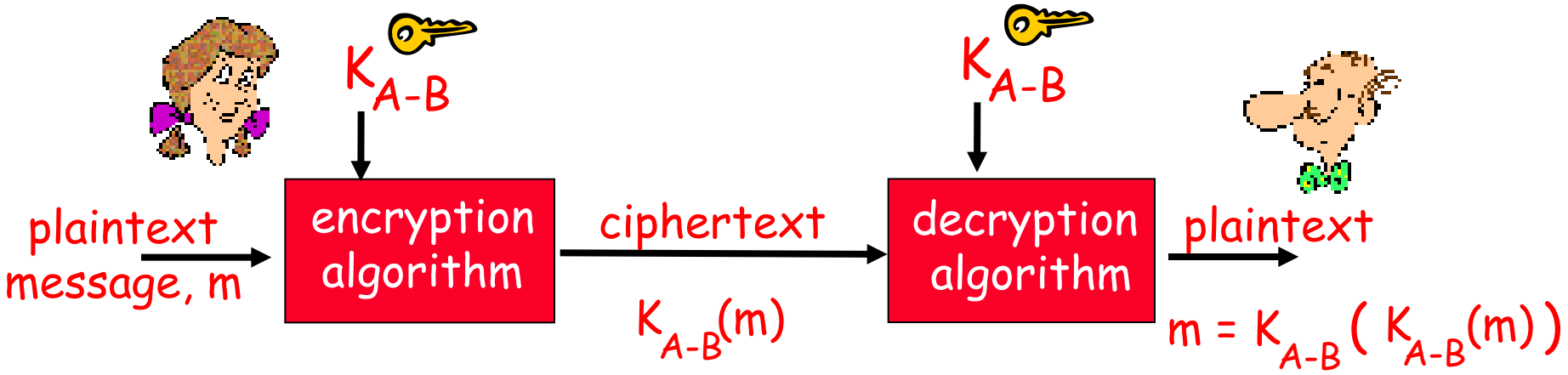
Beide Schlüssel sind identisch – **Shared Secret**

Asymmetrische Verschlüsselung:

Paar aus **öffentlichem** und **privatem** Schlüssel

(Public Key, Private Key), (Privater Schlüssel ist geheim)

Symmetrische Verschlüsselung



Symmetrische Verschlüsselung:

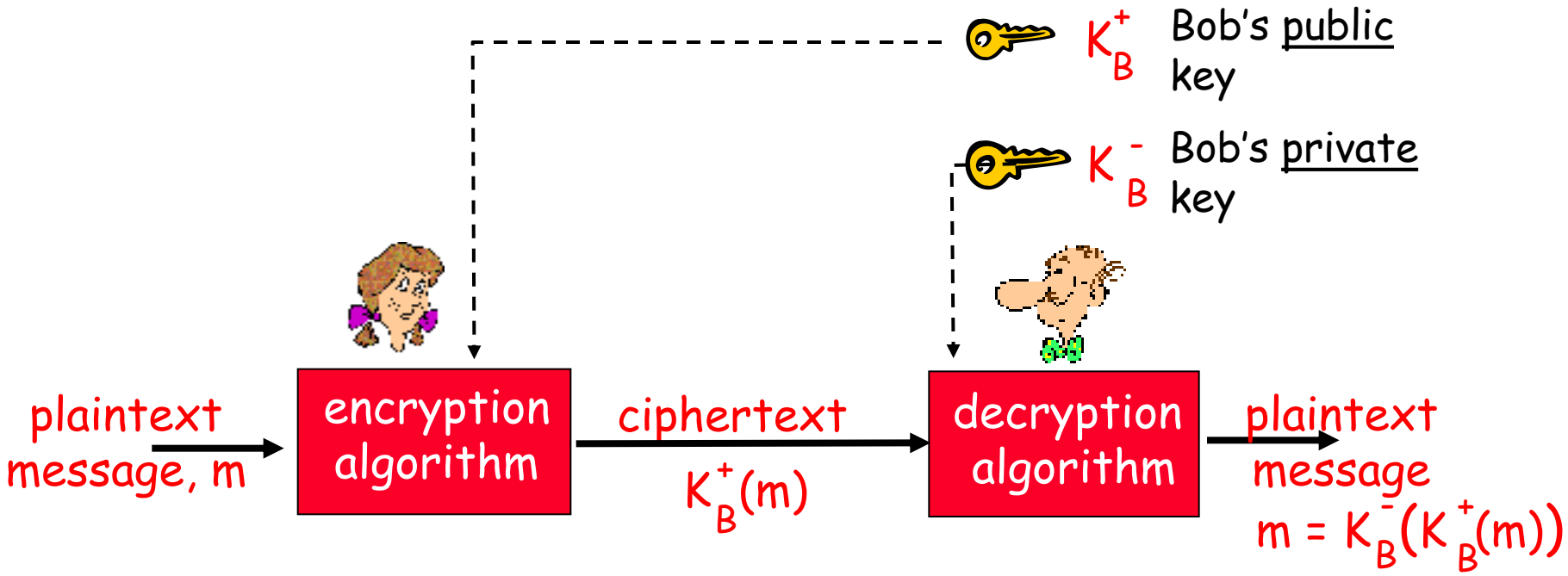
Bob and Alice kennen beide gemeinsam denselben Schlüssel: Shared Secret K_{A-B}

- Problem

  Das Shared Secret muss irgendwann vorher einmal auf sichere Weise kommuniziert worden sein: *Man kann nur dann sicher kommunizieren, wenn man vorher schon einmal sicher kommunizieren konnte!*
- Vorteil

  Leistungsfähige Algorithmen und Implementierungen verfügbar.
- Beispiele: DES, TripleDES, AES

Public Key Kryptographie – Asymmetrische Verschlüsselung



Public Key Kryptographie [Diffie-Hellman76, RSA78]

- Es gibt kein geteiltes Geheimnis
- *Alle kennen den öffentlichen Schlüssel*
- *Nur der Empfänger kennt den privaten Entschlüsselungsschlüssel*

Authentifikation

Bob und Alice kommunizieren per Nachrichtenaustausch.

Ziel: Bob möchte, dass Alice ihm beweist, dass sie wirklich Alice ist

Protokoll ap1.0: Alice teilt mit "Ich bin Alice"

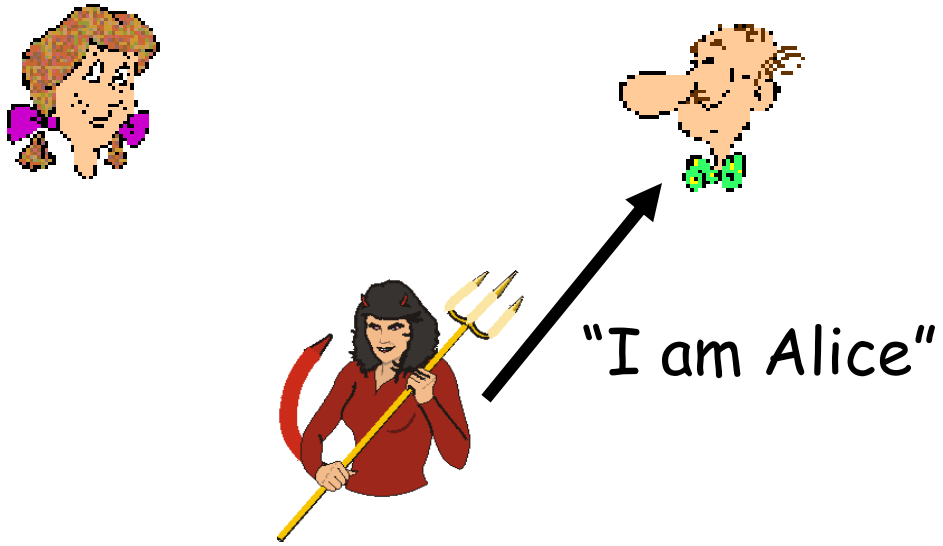
Fehlermöglichkeiten??

Authentifikation

Bob und Alice kommunizieren per Nachrichtenaustausch.

Ziel: Bob möchte, dass Alice ihm beweist, dass sie wirklich Alice ist

Protokoll ap1.0: Alice teilt mit "Ich bin Alice"

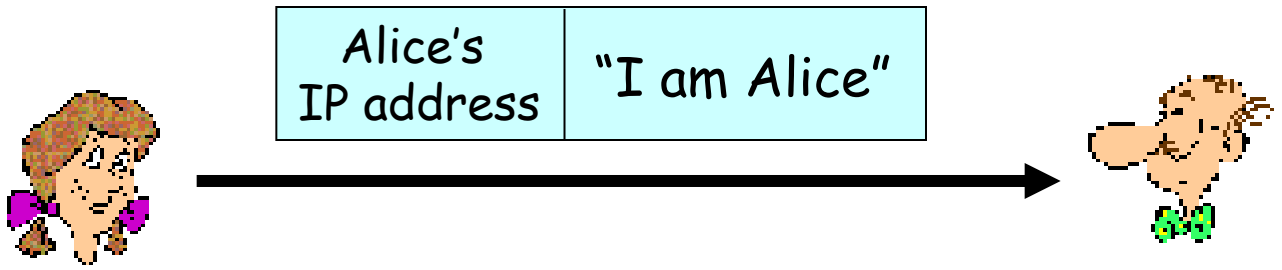


Da Bob Alice nicht sehen kann, kann Trudy einfach behaupten, selbst Alice zu sein

Authentifikation

Protokoll ap2.0:

Alice teilt per IP-Paket mit ihrer IP-Adresse als Absenderadresse mit "Ich bin Alice"

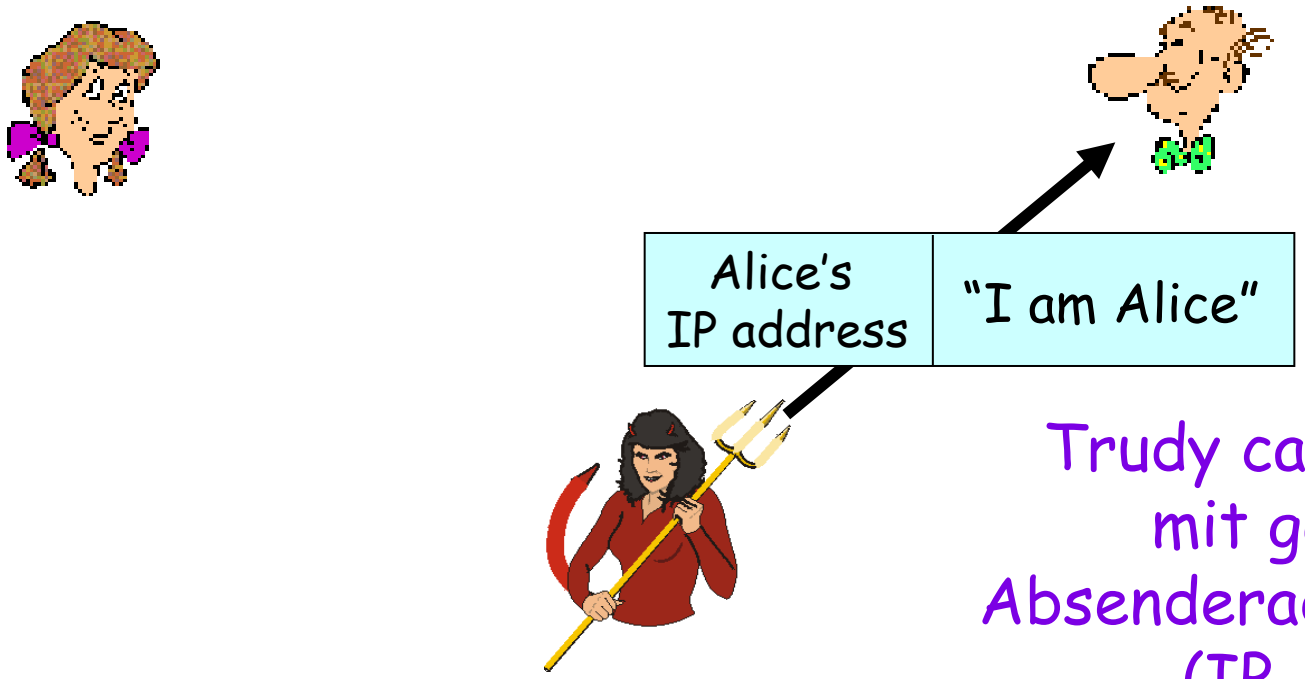


Fehlermöglichkeiten??

Authentifikation

Protokoll ap2.0:

Alice teilt per IP-Paket mit ihrer IP-Adresse als Absenderadresse mit "Ich bin Alice"

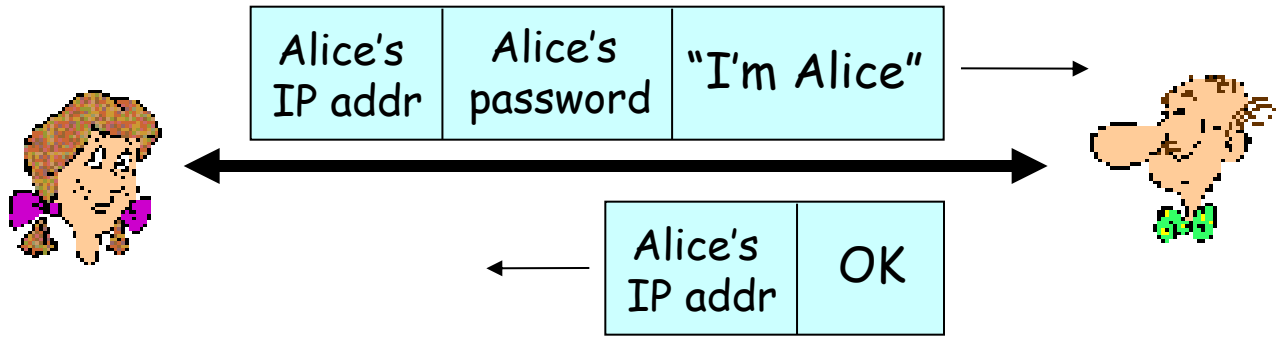


Trudy can ein IP-Paket mit gefälschter Absenderadresse erzeugen (IP-Spoofing)

Authentifikation

Protokoll ap3.0:

Alice teilt mit "Ich bin Alice" und sendet ihr geheimes Passwort als Beweis mit

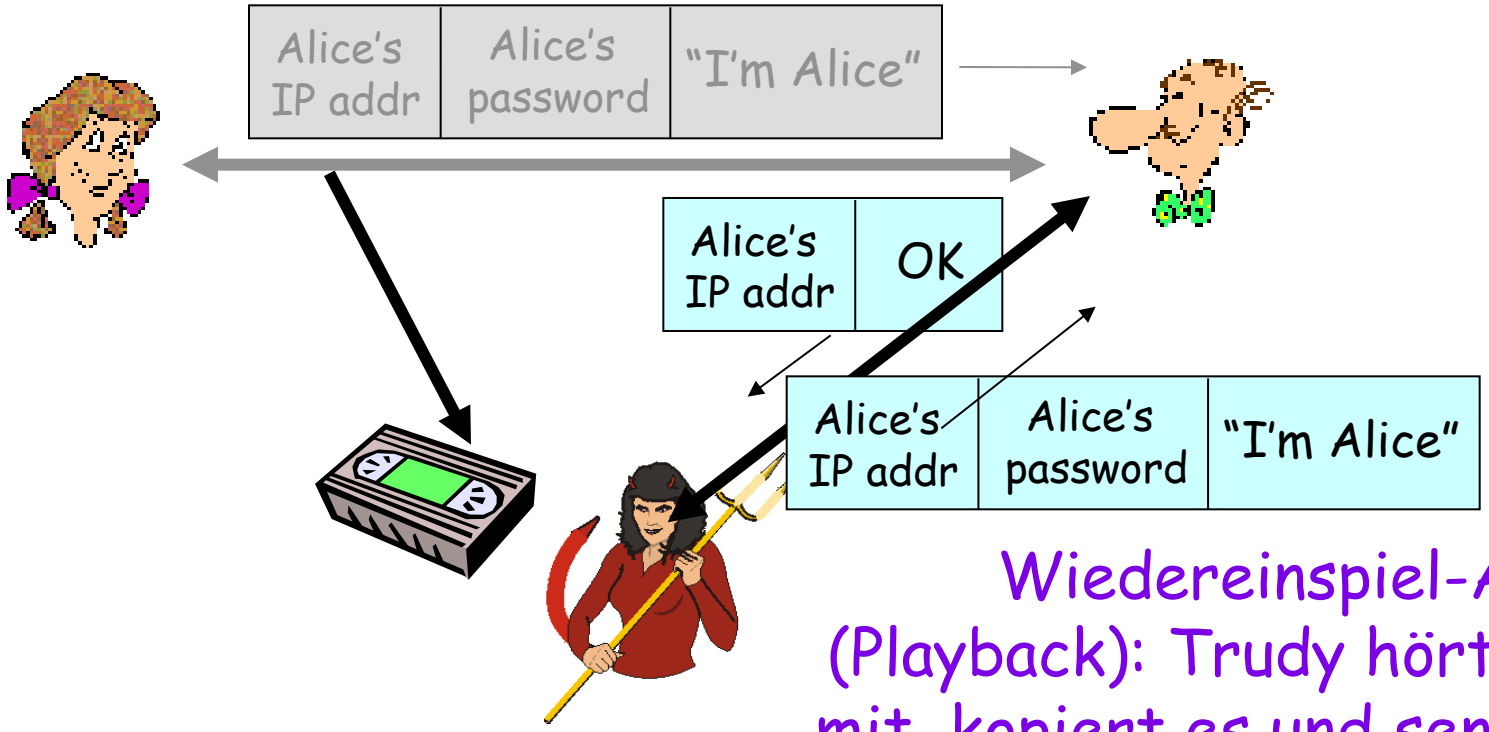


Schwachstellen??

Authentifikation

Protokoll ap3.0:

Alice teilt mit "Ich bin Alice" und sendet ihr geheimes Passwort als Beweis mit

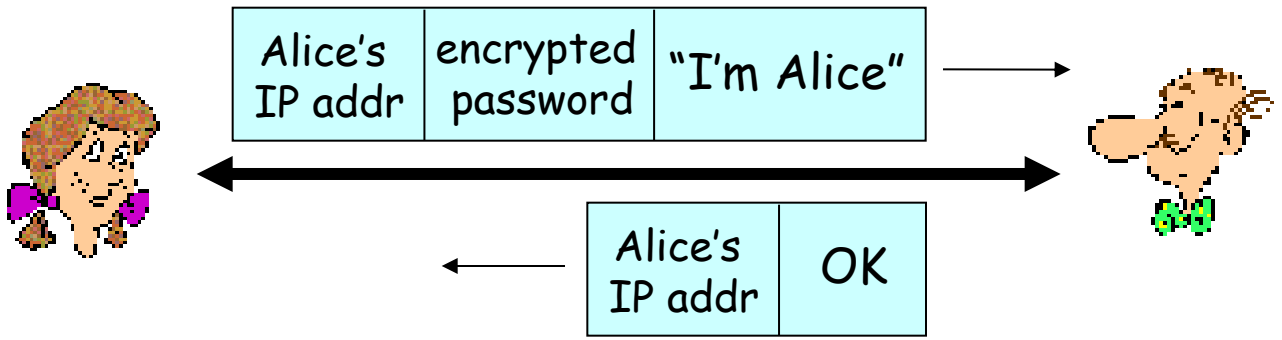


Wiedereinspiel-Attacke (Playback): Trudy hört Alices Paket mit, kopiert es und sendet es später an Bob

Authentifikation

Protokoll ap3.1:

Alice teilt mit "Ich bin Alice" und sendet ihr geheimes Passwort in verschlüsselter Form als Beweis mit

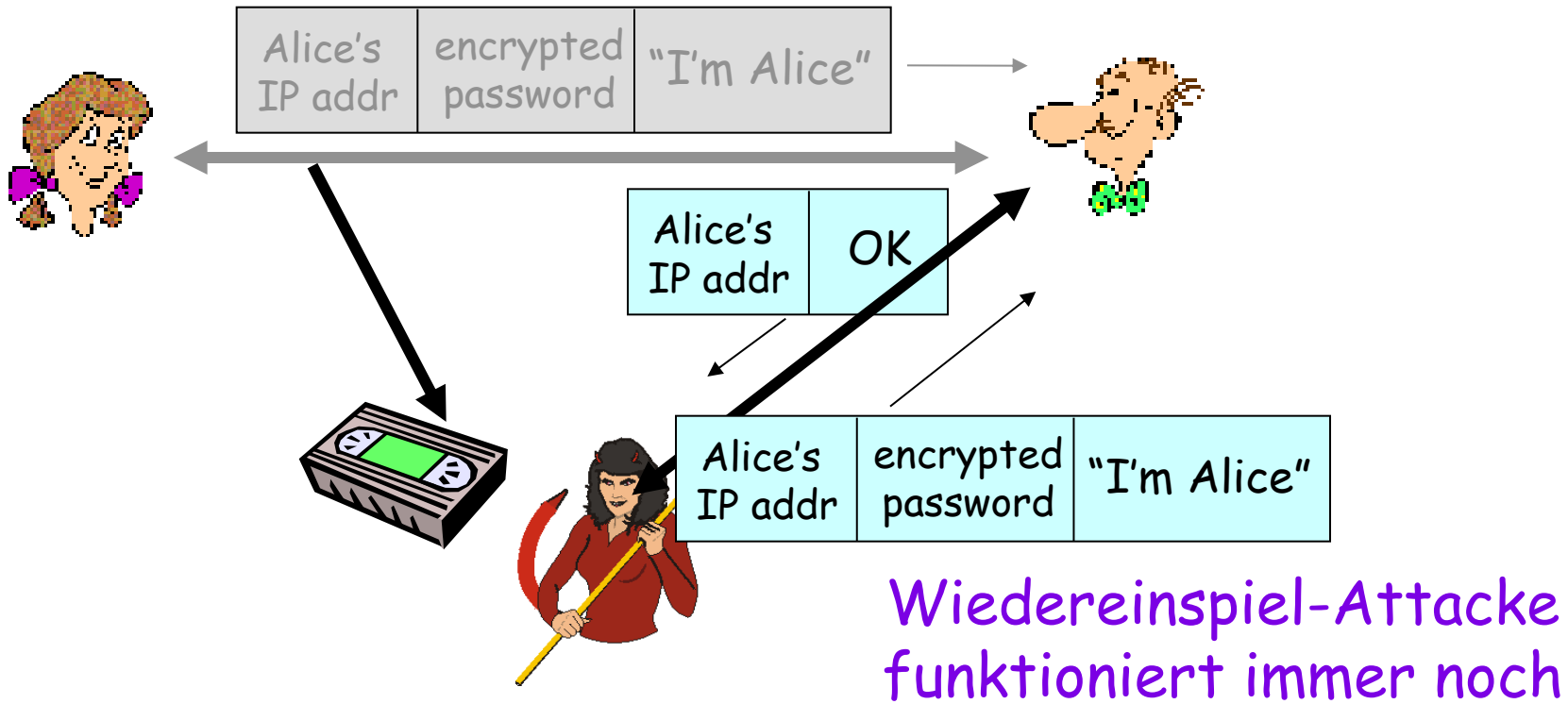


Schwachstellen??

Authentifikation

Protokoll ap3.1:

Alice teilt mit "Ich bin Alice" und sendet ihr geheimes Passwort in verschlüsselter Form als Beweis mit

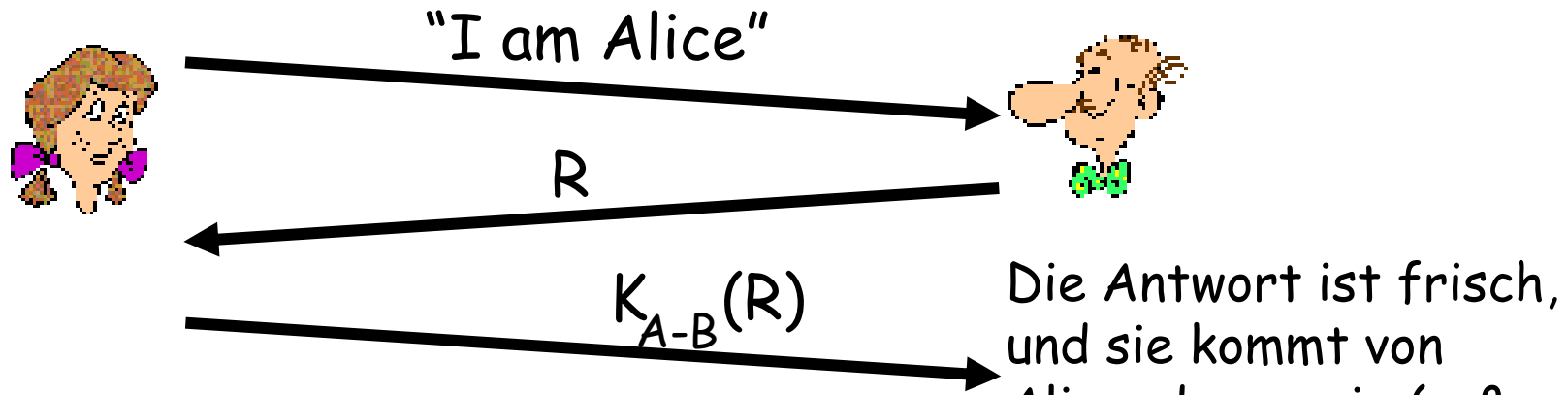


Authentifikation: Nächster Versuch

Ziel: Verhindere erfolgreiche Playback-Attacken

Nonce: Zahl, die nicht vorhersagbar ist und nur einmal benutzt wird (N_{once})

ap4.0: Als Beweis dafür, dass Alices Antwort "frisch" ist, sendet Bob eine Nonce R an Alice, Alice muss R in verschlüsselter Weise zurücksenden (Challenge-Response-Authentifkation)



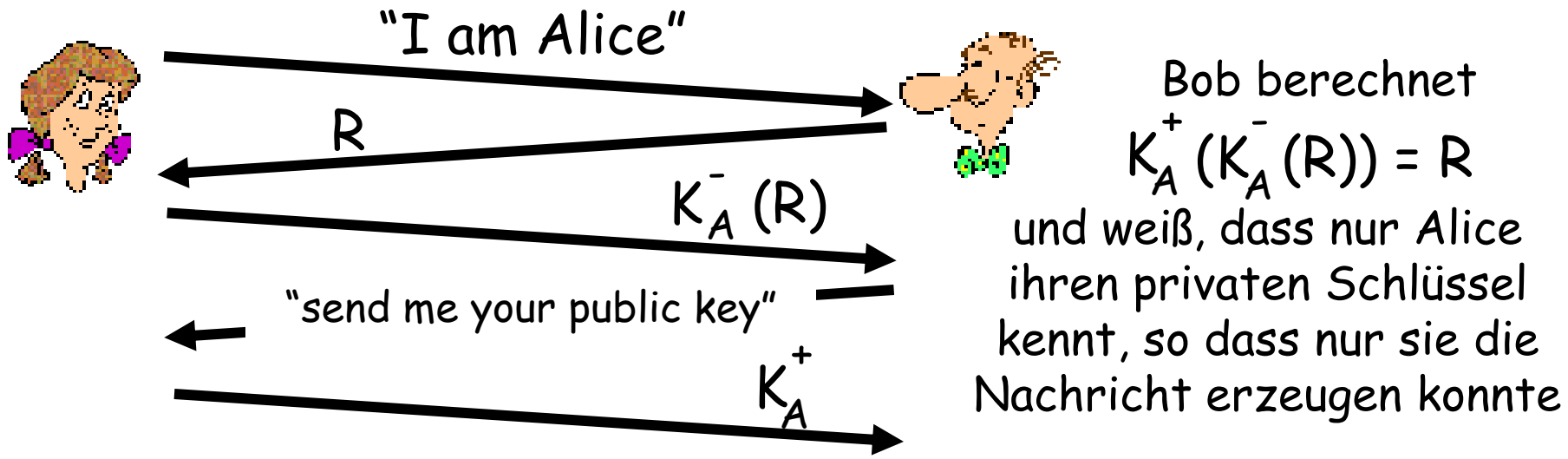
Schwachstellen??
Achtung Alice:
Bob hat sich nicht authentifiziert!

Authentifikation mit Public Key Kryptographie

ap4.0 benötigt ein Shared Secret K_{A-B} , das initial beiden bekannt sein muss

- Geht es auch mit Public-Key-Verschlüsselung?

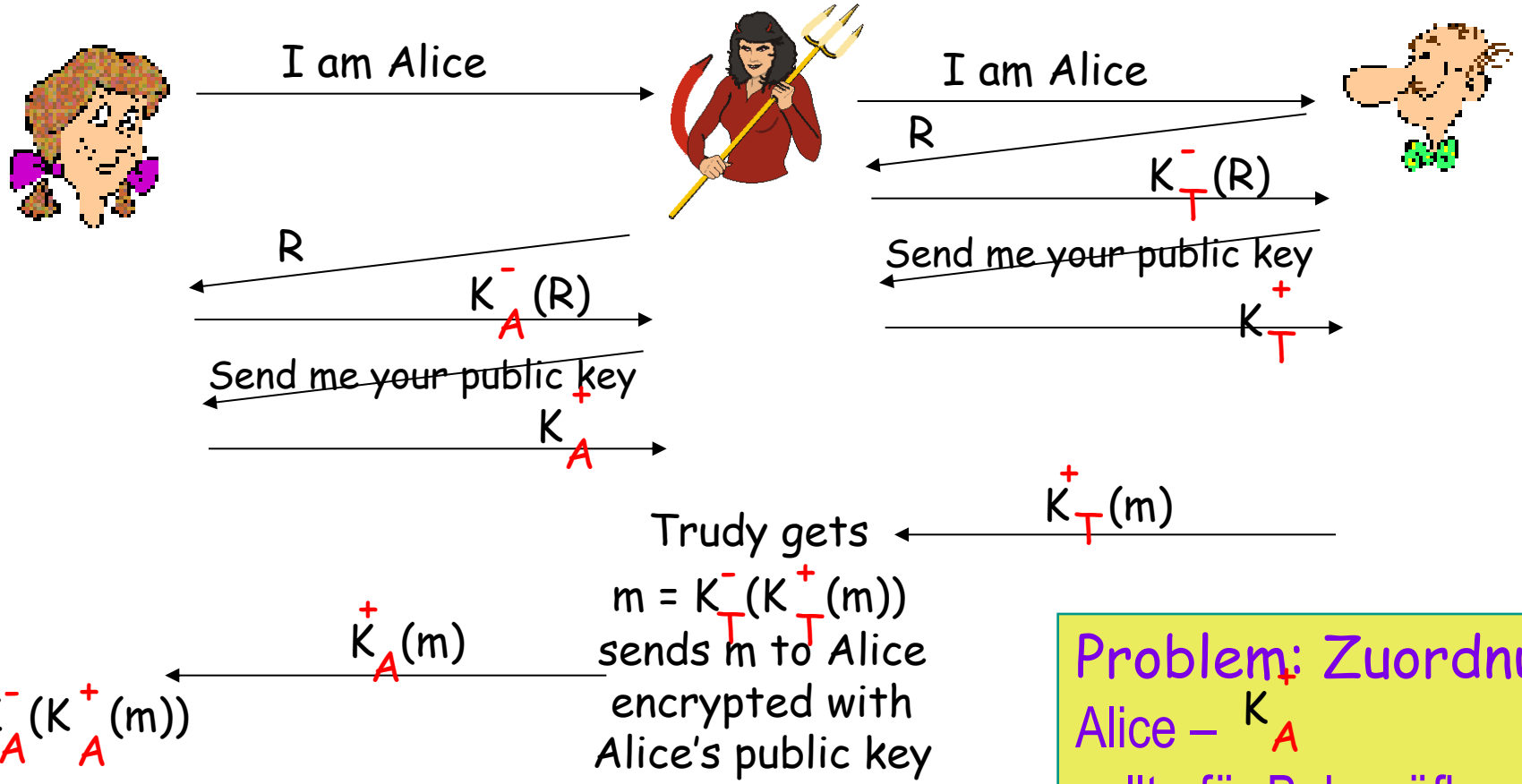
ap5.0: Nonce und Signatur



ap5.0: Schwachstelle – "Man in the Middle" Angriff

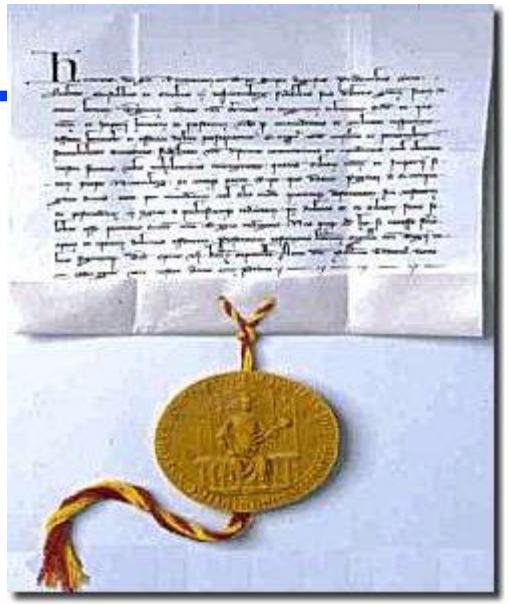
Man (woman) in the middle attack:

Trudy gibt sich bei Bob als Alice und bei Alice als Bob aus



Problem: Zuordnung Alice – K_A^+ sollte für Bob prüfbar sein

Digitale Unterschrift (Digital Signature)



Kryptographische Technik, welche die Funktion handschriftlicher Unterschriften erfüllen soll

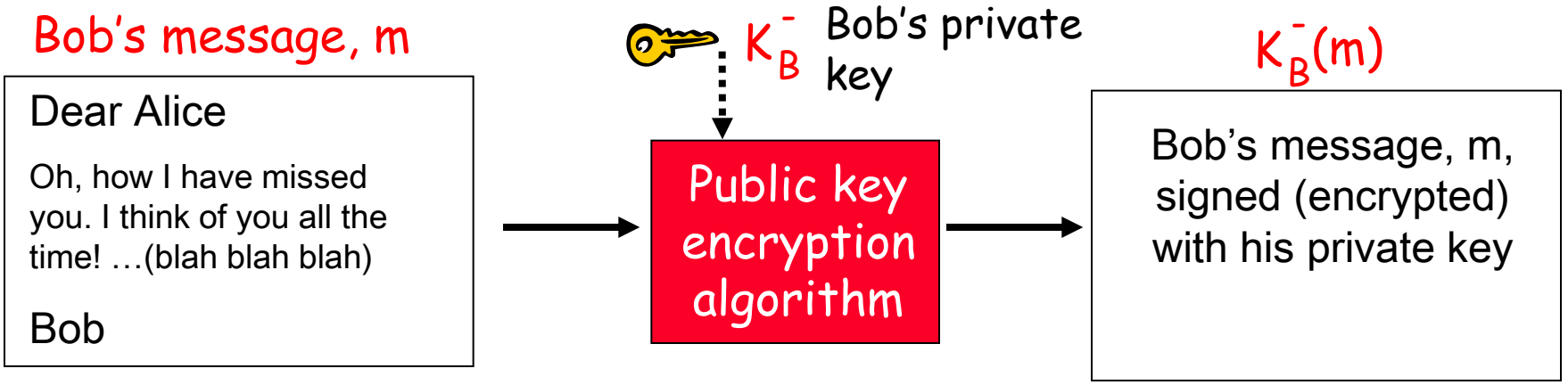
- Sender (Bob) signiert ein Dokument digital und bestätigt damit, dass er das Dokument so erzeugt hat
- verifizierbar, fälschungssicher:
  Empfänger (Alice) kann Dritten gegenüber beweisen, dass Bob, und niemand anders (auch Alice nicht), das Doklument signiert haben muss

- ABER:
  - Kryptoalgorithmen sind nicht ewig sicher:
    Digitale Unterschriften müssen alle paar Jahre aufgefrischt werden
  - Private Schlüssel können korrumpiert werden: Rückrufe

Digitale Signatur

Einfache digitale Signatur für eine Nachricht m:

- Bob signiert m dadurch, dass er m mit seinem privaten Schlüssel K_B^- verschlüsselt: $K_B^-(m)$



Wenn Alice diese Nachricht empfängt, den öffentlichen Schlüssel von Bob kennt und davon ausgehen kann, dass Bobs privater Schlüssel nur Bob bekannt ist:

• Bob und kein anderer hat diese Nachricht so signiert

• Bob kann nicht abstreiten, dass er die Nachricht signiert hat

Probleme:

• Asymmetrische Verschlüsselung ist rechenaufwendig

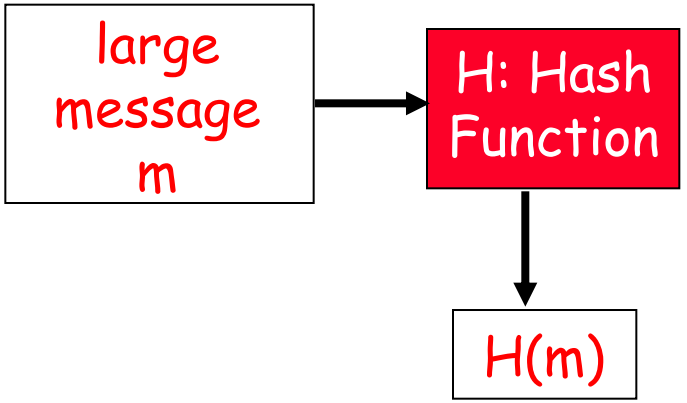
• Wie erfährt Alice den öffentlichen Schlüssel K_B^+ von Bob?

Message Digest – Kryptographische Hashfunktion

Das direkte Signieren langer Nachrichten kostet viel Rechenzeit

Ziel: effizient berechenbarer Fingerabdruck einer Nachricht m: Message Digest H(m)

- H ist kryptographische Hashfunktion



- *Beispiele*
  MD5 (RFC 1321)
  - computes 128-bit message digest in 4-step process.
  - arbitrary 128-bit string x, appears difficult to construct msg m whose MD5 hash is equal to x.

  SHA-1 (NIST Standard)

Eigenschaften kryptographischer Hashfunktionen:

- Abbildung langer Bytefolgen auf kürzere Folge
- Nicht umkehrbar:
  Gegeben x = H(m), so ist es allzu aufwendig daraus m zu berechnen
- Gegeben m und x=H(m), so ist es allzu aufwendig ein m'≠m zu finden, so dass x=H(m') gilt.
- Es ist allzu aufwendig, überhaupt zwei m, m' zu finden, so dass H(m)=H(m') gilt

Internet Checksum: Zu schwach um Kryptohashfunktion zu sein

Internet Checksum hat einige Hashfunktionseigenschaften:

- ✓ Abbildung auf kurze Bytefolge
- ✓ Streuung

Aber, es ist sehr leicht, zu einer Nachricht m eine andere Nachricht m' zu finden, welche denselben Funktionswert hat:

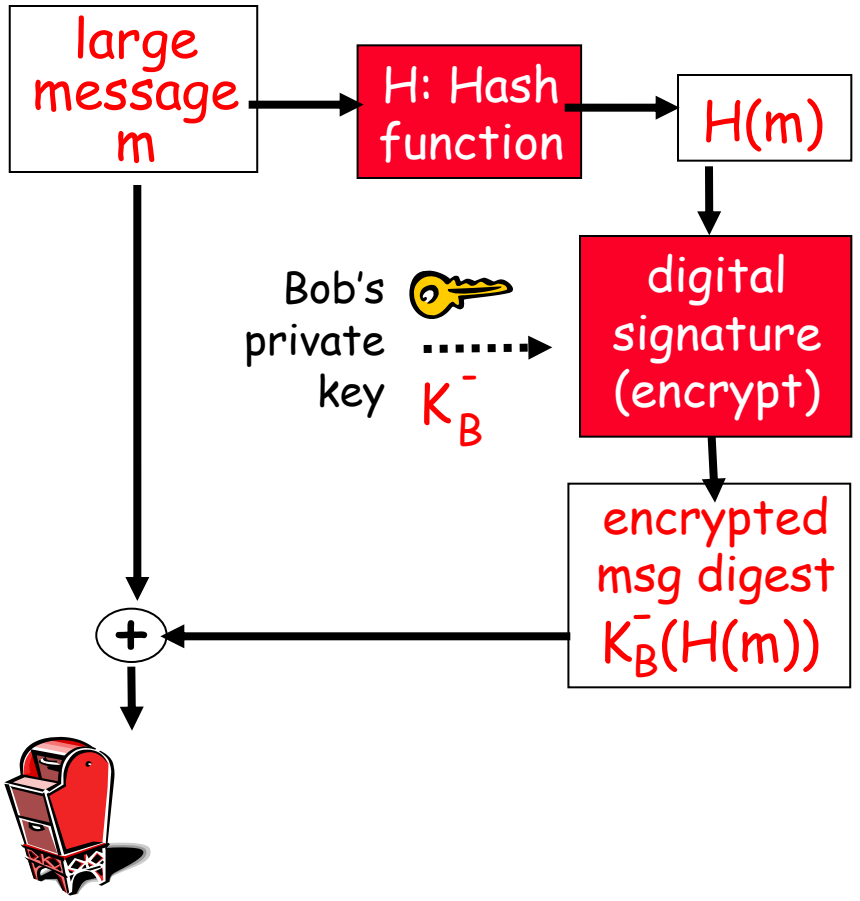
message	ASCII format
I O U 1	49 4F 55 31
0 0 . 9	30 30 2E 39
9 B O B	39 42 D2 42
	B2 C1 D2 AC

message	ASCII format
I O U 9	49 4F 55 39
0 0 . 1	30 30 2E 31
9 B O B	39 42 D2 42
	B2 C1 D2 AC

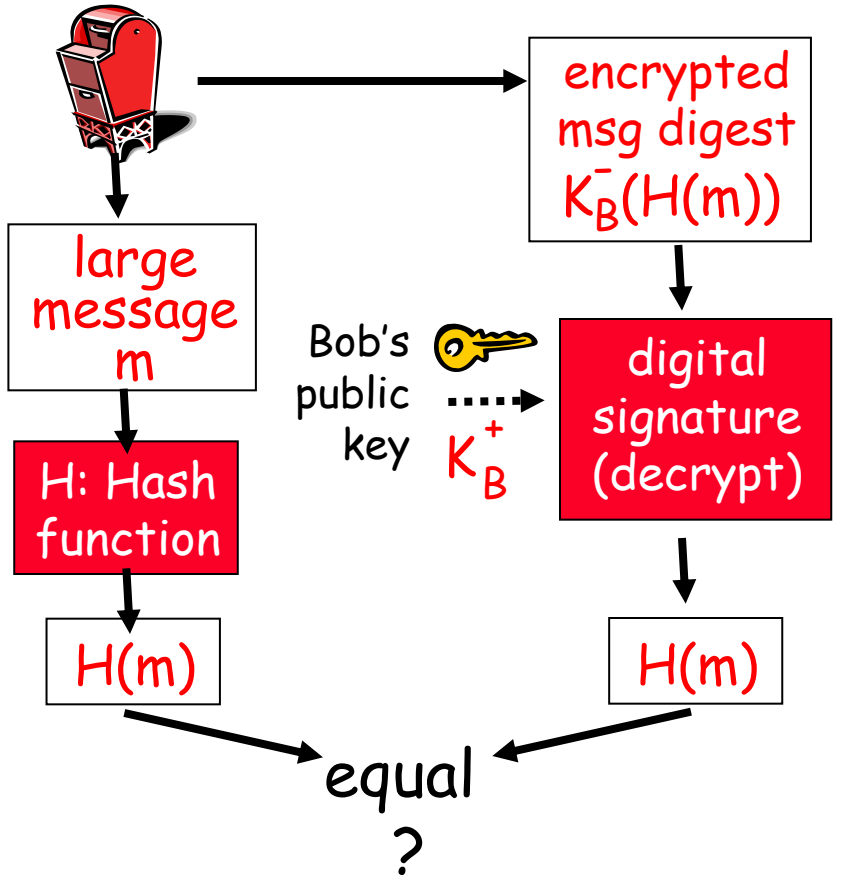
Verschiedene Nachrichten aber gleiche Prüfsummen!

Digitale Signatur = Signierter Message Digest

Bob sendet digital signierte Nachricht



Alice verifiziert die Signatur und die Integrität der signierten Nachricht



Vertrauenswürdige dritte Parteien

Verwaltung symmetrischer Schlüssel:

- Wie können 2 Parteien im Netz ein Shared Secret etablieren?

Lösung:

- Key Distribution Center (KDC) wirkt als Mittler zwischen den Parteien
  - statt n^2 Shared Secrets zwischen allen Paaren sind initial nur n Shared Secrets zwischen KDC und den Parteien einzurichten
  - KDC generiert bei Bedarf Sitzungsschlüssel für 2 Parteien

Public Key Zertifizierung:

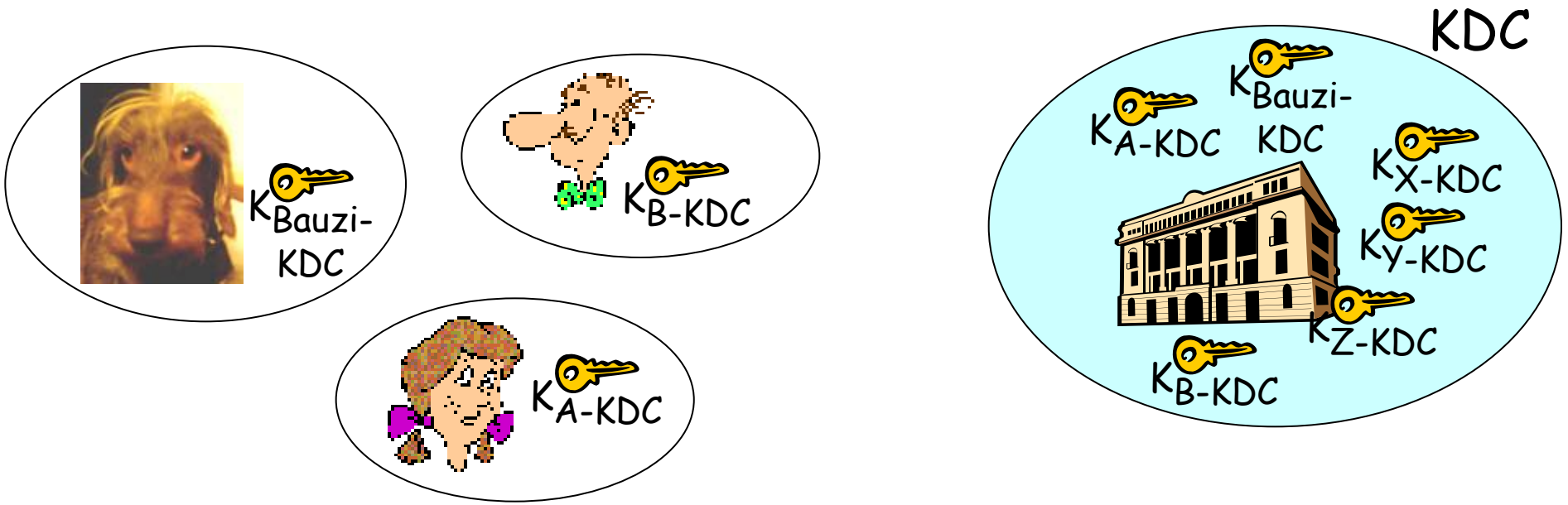
- Wenn Alice den öffentlichen Schlüssel von Bob erfährt, wie kann sie sicher sein, dass das wirklich Bobs öffentlicher Schlüssel ist

Lösung:

- Zertifizierungsstelle (Certification Authority CA)

Key Distribution Center (KDC)

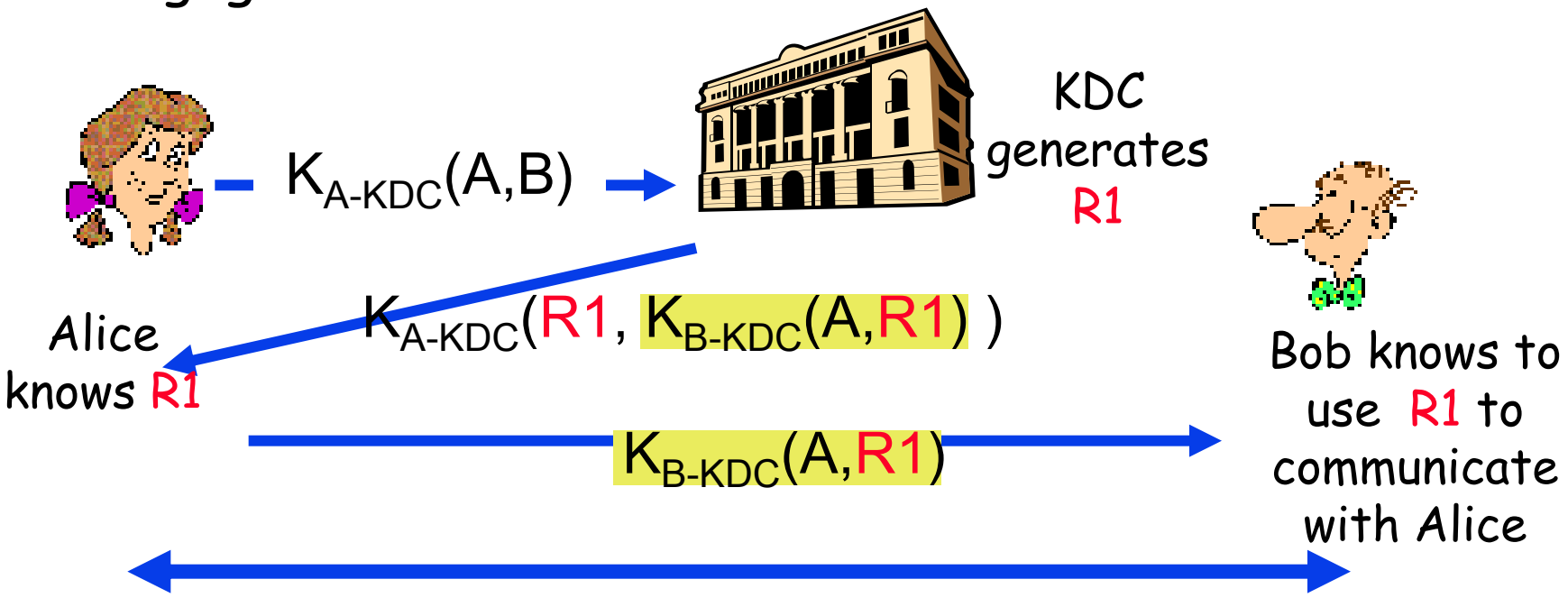
- Alice, Bob brauchen ein Shared Secret zur effizienten sicheren Kommunikation
- KDC: Server verwaltet je Partei einen geheimen Schlüssel
- Alice und Bob kennen jeweils ihre eigenen geheimen Schlüssel, $K_{A\text{-}KDC}$ $K_{B\text{-}KDC}$, mit deren Hilfe sie mit dem KDC authentifiziert kommunizieren können.
- Wenn Alice eine Sitzung mit Bob durchführen will, lassen sie sich vom KDC einen Sitzungsschlüssel als Shared Secret zwischen Alice und Bob erzeugen



Key Distribution Center (KDC)

Wie erfährt Bob den Sitzungsschlüssel R1?

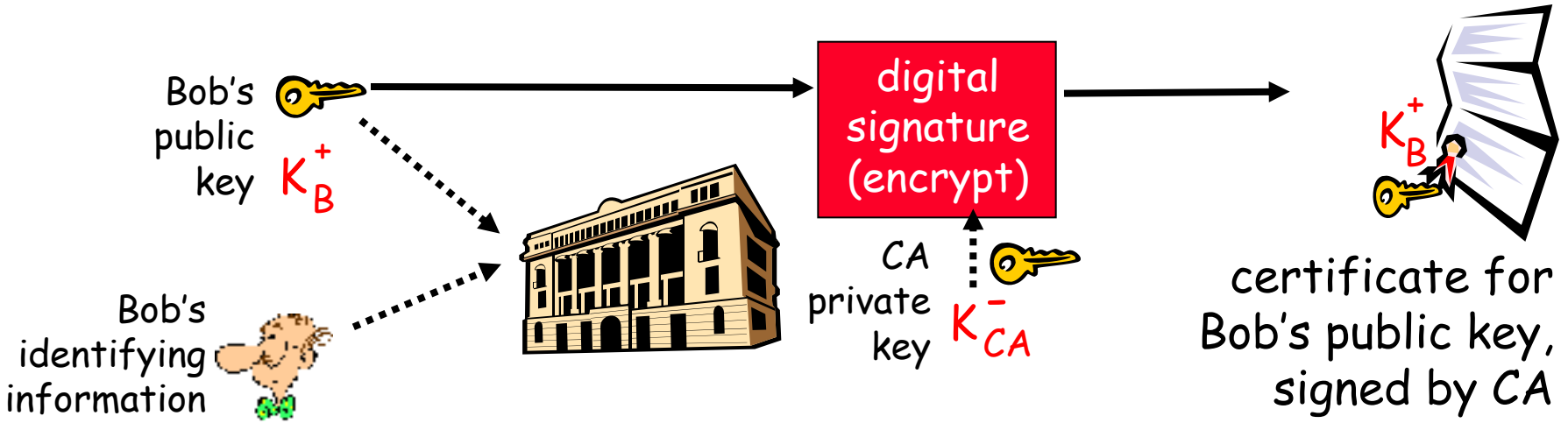
KDC erzeugt "Ticket", das von Alice unveränderbar an Bob weitergegeben wird



Alice und Bob kommunizieren effizient: Sie nutzen R1 als *Session Key* für die symmetrische Verschlüsselung

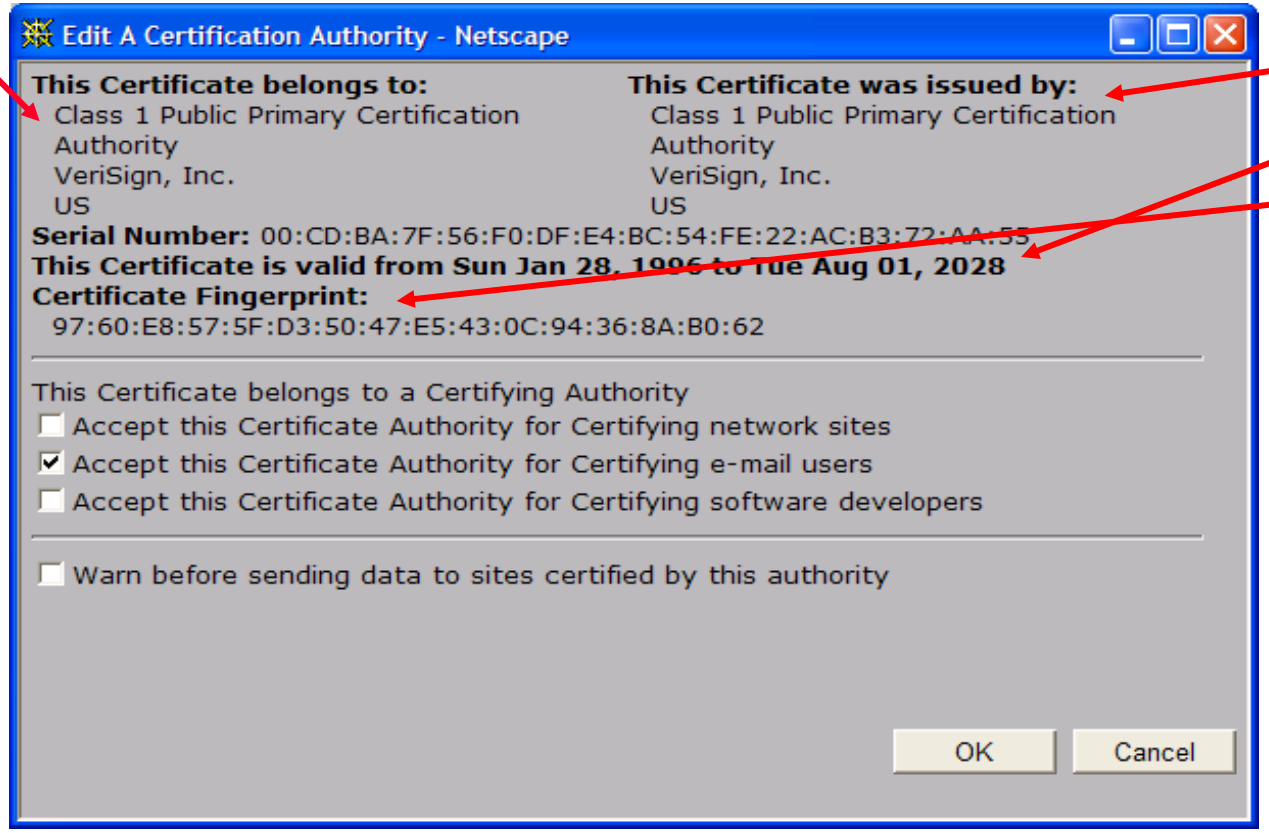
Zertifizierungsstellen (Certification Authorities CAs)

- Certification Authority (CA): Verwalte die Bindung eines öffentlichen Schlüssels an Person / Partei E.
- E registriert seinen öffentlichen Schlüssel bei CA.
  - E weist sich bei CA aus (z.B. mit dem Personalausweis)
  - CA erzeugt einen Datensatz, das Zertifikat, das die Bindung von K_E^+ an E dokumentiert
  - Zertifikat: "K_E^+ ist öffentlicher Schlüssel von E" digital signiert von CA



Inhalt eines Zertifikats

- Seriennummer (eindeutig für alle Zertifikate derselben CA)
- Information zur Partei: Name, Art
  - auch (hier nicht sichtbar) öffentlicher Schlüssel sowie Angaben zu unterstützten Kryptoalgorithmen



Info zu CA

Gültigkeitszeitdauer

Signatur der CA

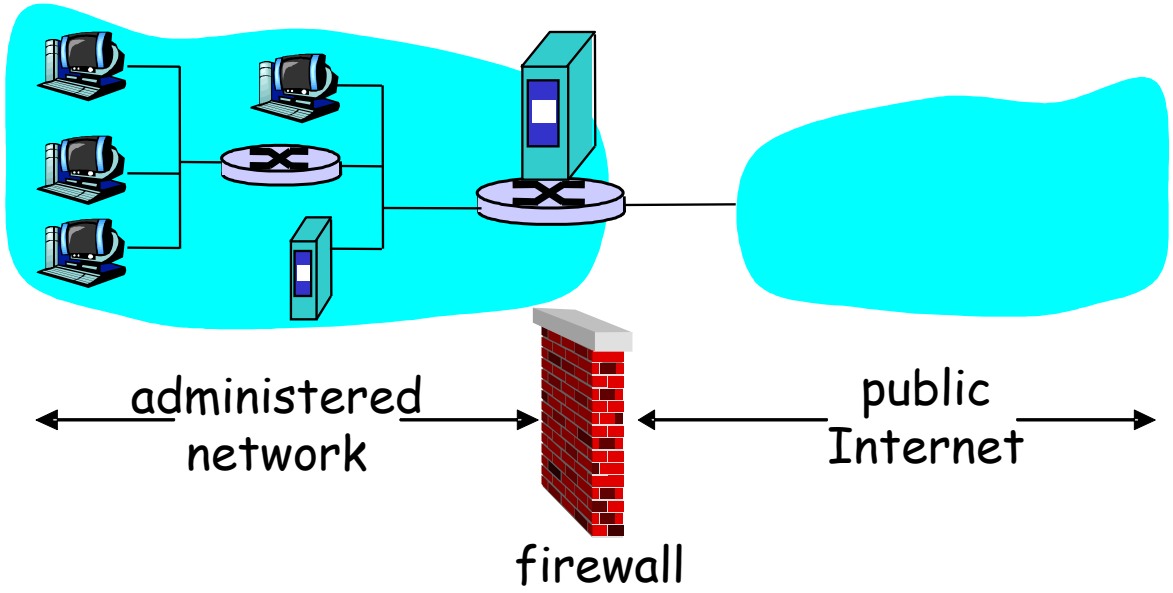
Weitere Aufgaben einer CA

- Zeitstempel
- Rückruf-Listen

Firewalls

Firewall

Verkehrskontrolleinrichtung an Grenze eines Firmennetzes zum öffentlichen Netz hin (auch an Innennetzgrenzen zu sensiblen Subnetzen): Lässt manche Kommunikation zu, manche nicht.



Firewalls: Motivation

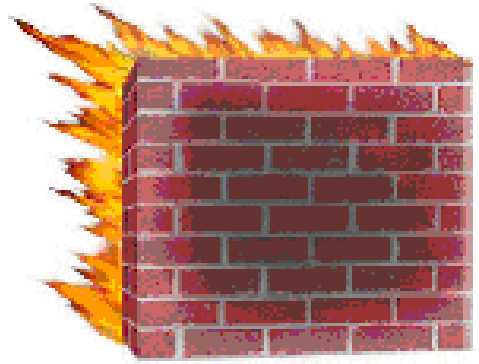
Eigentlich sind Firewalls nicht nötig, weil alle Hosts und Router nur vorgesehene Dienste an vorgesehene Nutzer erbringen sollen und dies durch die Autorisierungs- und Authentifikationsdienste der Rechner kontrolliert wird.

Aber es gibt immer wieder unvorhergesehene Schwachstellen, die aus Programmier- und Administrationsfehlern resultieren.

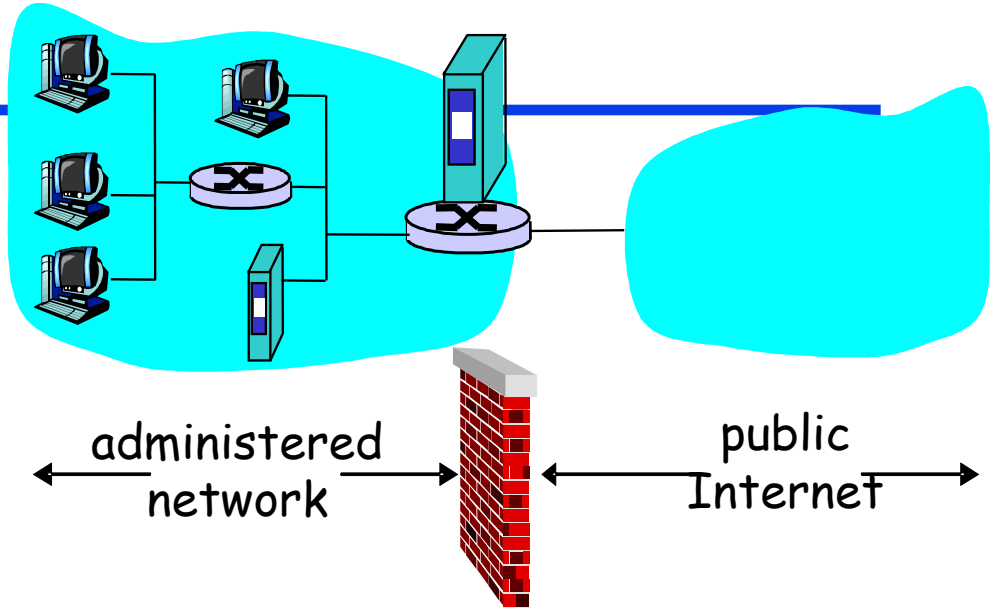
Deshalb sollen Firewalls zusätzlich unabhängig von den anderen Diensten unerwünschten Verkehr abblocken und damit die Angriffsfläche verkleinern.

Ferner

- Abwehr von Verfügbarkeitsangriffen auf das Innennetz
- Abwehr von IP-Spoofing-Angriffen
- Oft in Verbindung mit NAT
- Oft in Verbindung mit VPN



Firewalls: Architektur

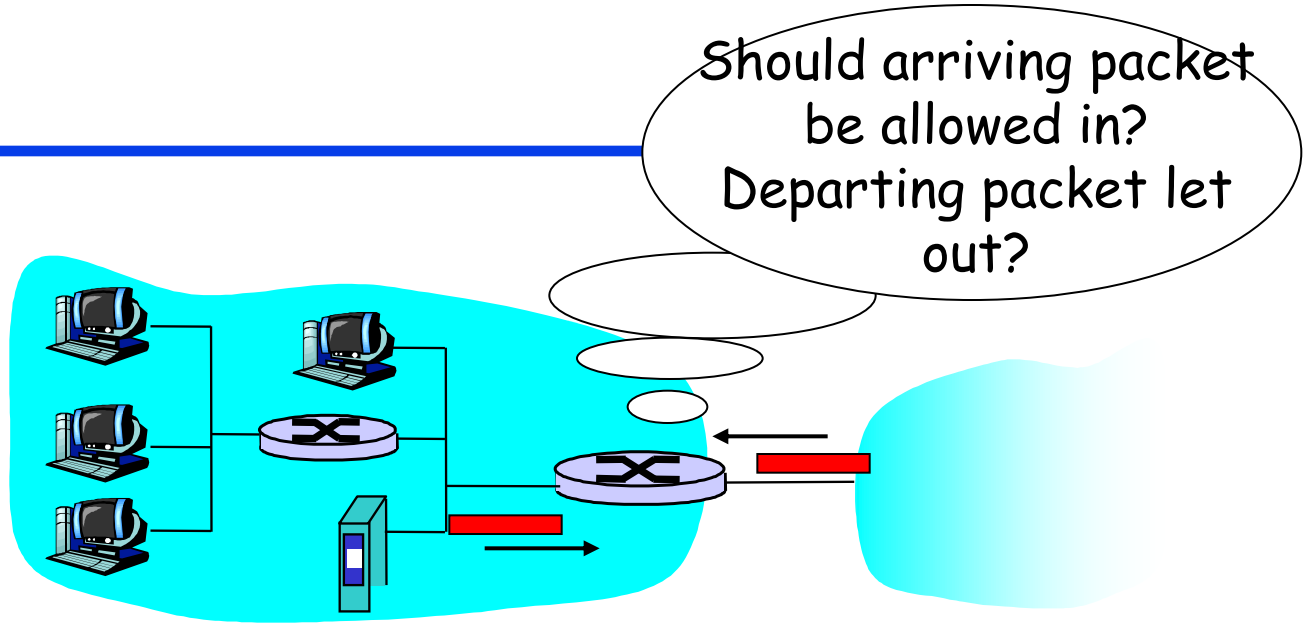


Drei Aspekte

- **Netztopologie**
  - **Innennetz – Außennetz, Firewall an Verbindungswegen**
- **Filterfunktion**
  3 Filtertypen
  - **Applikationsfilter**
  - **Verbindungsfilter**
  - **Paketfilter (statisch / dynamisch)**
- **Filteranordnung**
  - nur ein Router mit Paketfilter
  - mehrere zusammenwirkende Filter und Knoten
    - » Dual homed Bastion Host
    - » Screened Subnet

Paket-Filter

- Router, der Innen- und Außennetz verbindet, hat Paketfilterfunktion
- Liste aus Filterregeln der Form
  "Interface, Bedingung über Paket-Header, Aktion"
- Bedingung:
  - source IP address, destination IP address, TCP/UDP source and destination port numbers
  - ICMP message type, TCP SYN and ACK bits
- Aktion: Paket durchlassen, verwerfen (mit / ohne Alarm)
- Statische und dynamische Filter



Filterlisten – Aufbau

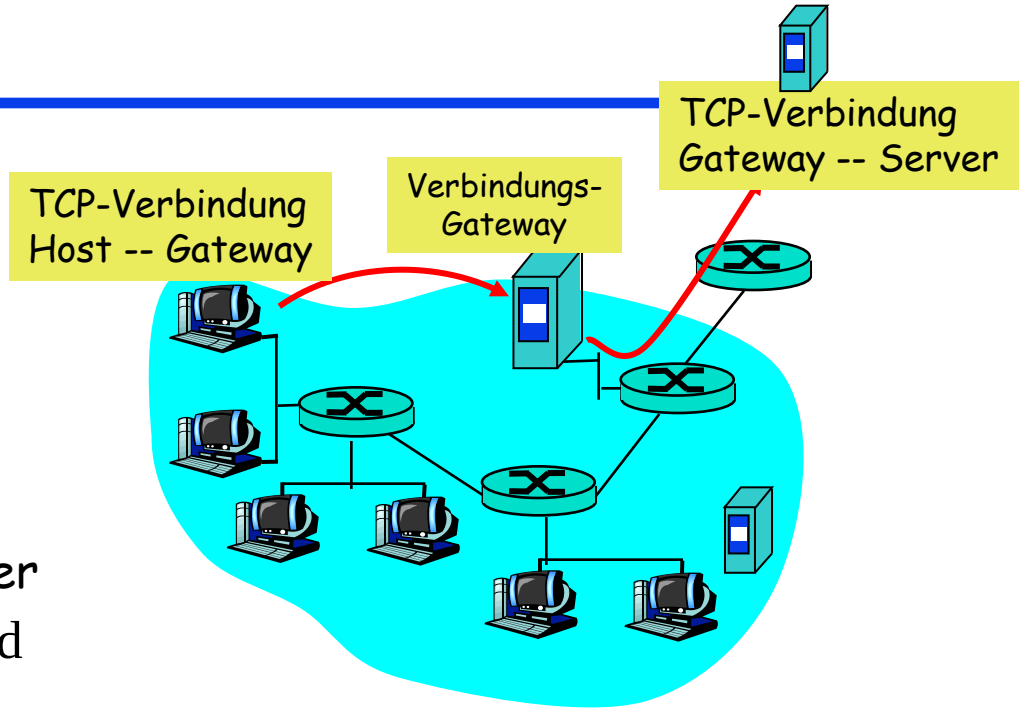
Vorne: Anti-Spoofing Regeln verbieten, dass von außen Pakete mit Innenadressen durchkommen

Mitte: Nur positive Regeln für den notwendigen Verkehr

Hinten: Negative Regeln, die den ganzen Rest verbieten.

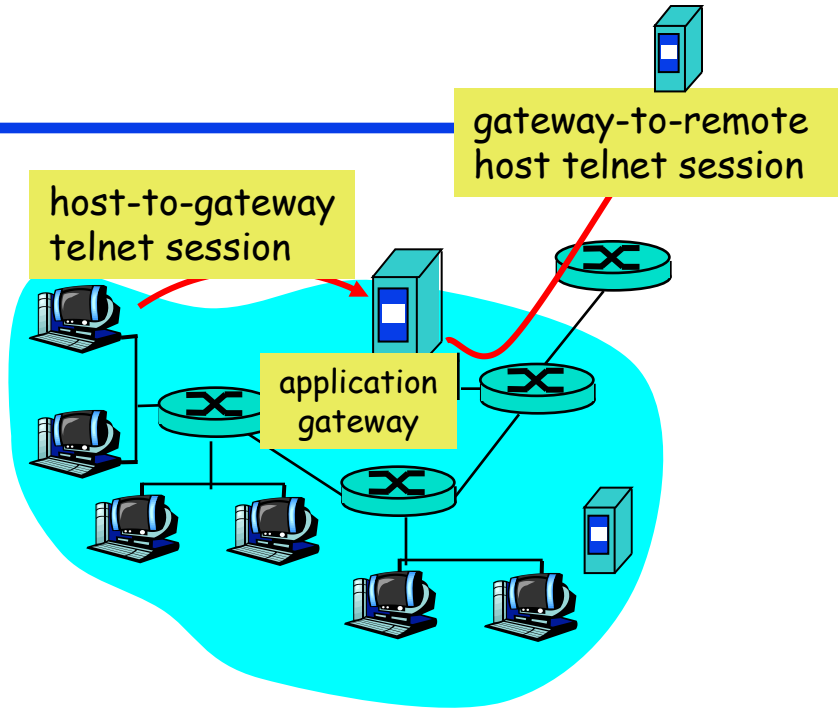
Verbindungsfilter

- Realisierung durch einen Prozess "**Verbindungs-Gateway**" auf einem Firewall-Host
- Es werden keine direkten Transportverbindungen mehr zwischen Außen- und Innennetz zugelassen:
  - Stattdessen 2 Verbindungen: **Client** – **Gateway** und **Gateway** – **Server**
- Gateway packt die TCP-Nutzdaten aus und verpackt sie selbst wieder
- Prüfung der TCP-Adressen und Formate, Erschweren von Formatfehler- und Segmentierungsattacken
- Die eigentlichen Anwendungsdaten können nicht untersucht werden, weil das Verbindungsgateway das Anwendungsprotokoll nicht kennt



Applikationsfilter

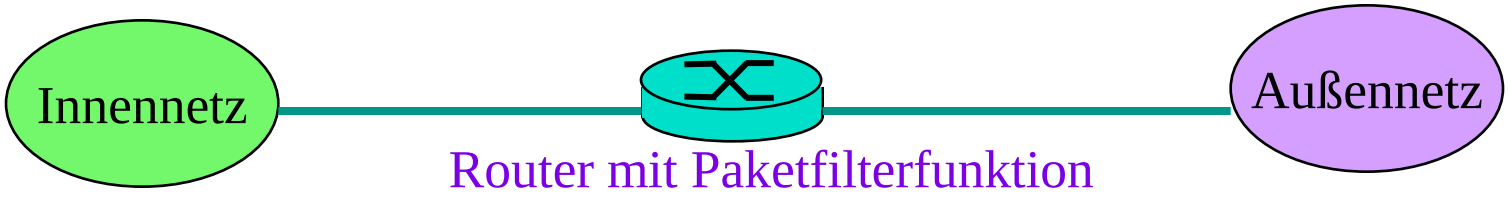
- Realisierung durch einen Prozess "**Applikationsgateway**" auf einem Firewall-Host, z.B. **Telnet-Gateway**
- Es werden keine direkten Anwendungsverbindungen mehr zwischen Außen- und Innennetz zugelassen:
  - Stattdessen 2 Verbindungen: **Client** – **Gateway** und **Gateway** – **Server**
- Gateway packt die Anwendungsnutzdaten aus und verpackt sie selbst wieder
- Gateway kann Anwendungsdaten interpretieren, da speziell für bestimmten Anwendungstyp erzeugt:
  - Nutzerkennungen, Authentifikation und Autorisierung
  - Zusatzdaten (z.B. Mail-Anhänge, Active X, Applets)



Ein Applikationsgateway wird oft auch Applikations-Proxy oder Applikationsfilter genannt

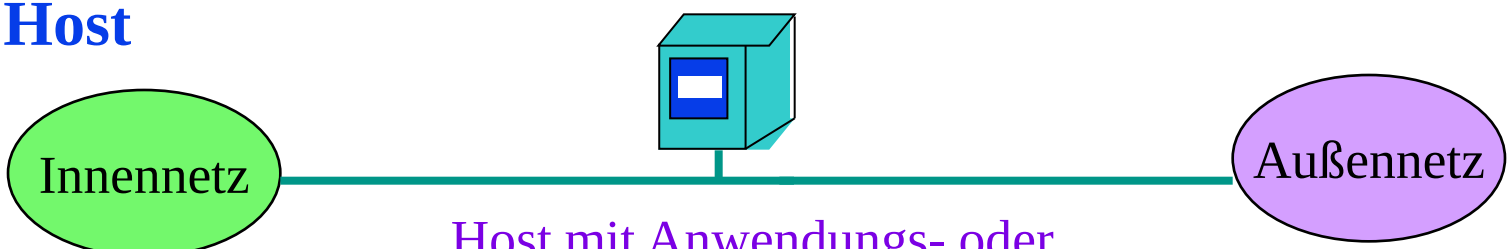
Firewall – Filteranordnung

Screening Router



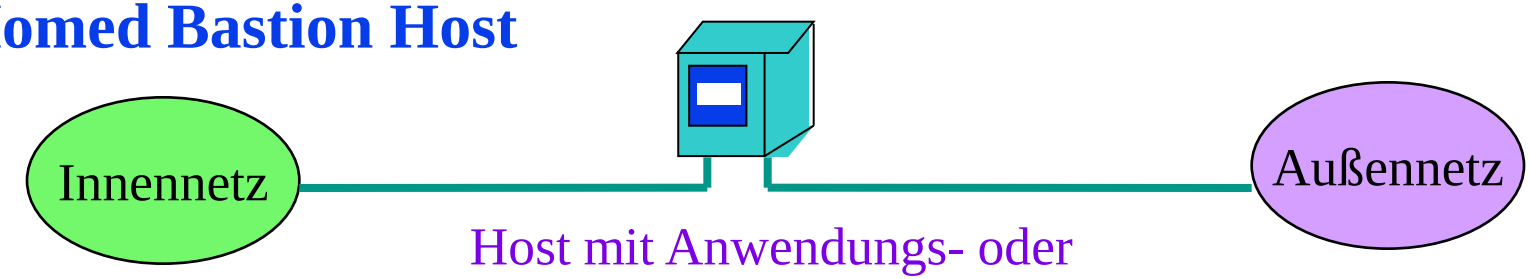
Router mit Paketfilterfunktion

Bastion Host



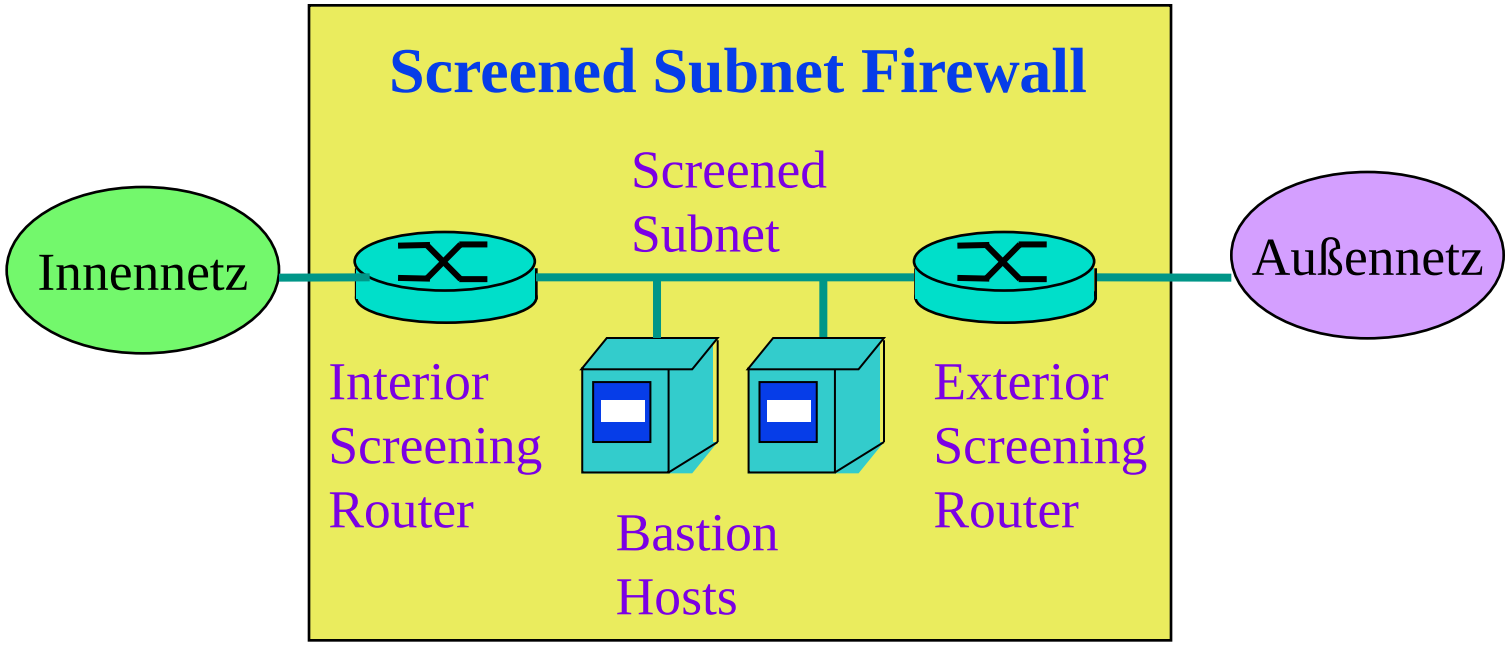
Host mit Anwendungs- oder
Verbindungsfiltern

Dual Homed Bastion Host



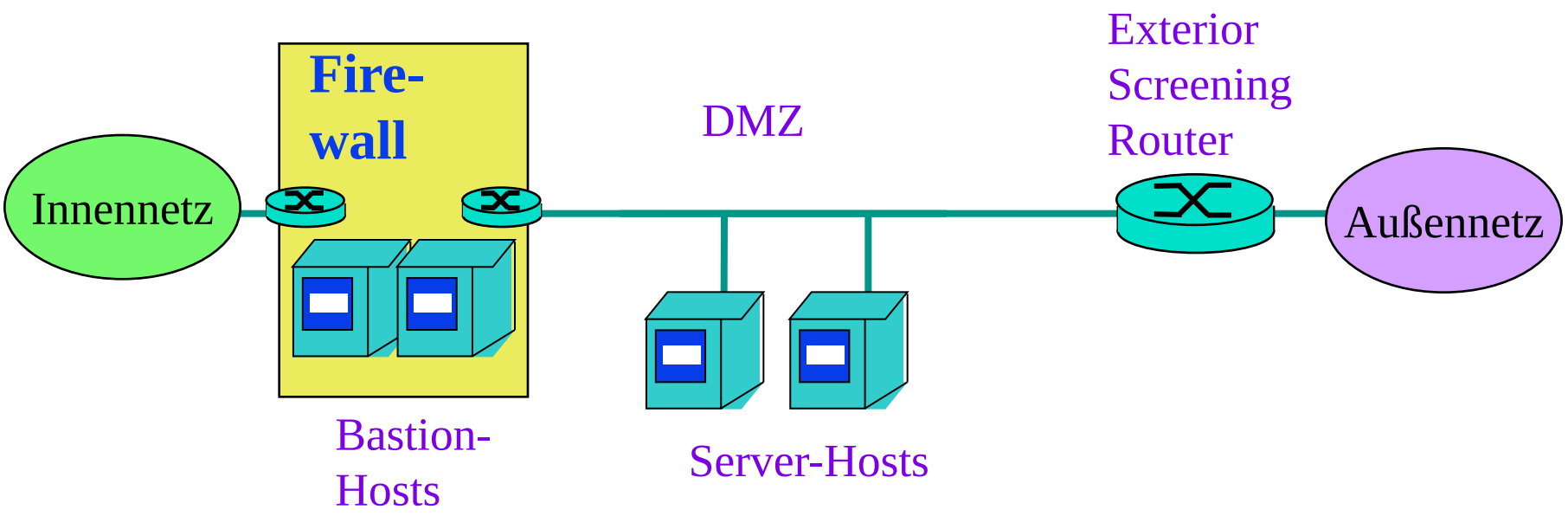
Host mit Anwendungs- oder
Verbindungsfiltern und 2 Netzinterfaces

Firewall – Filteranordnung



- Firewall besteht aus 2 Paketfiltern und einigen Bastion Hosts
  - Paketfilter schützen die Bastion Hosts und erzwingen, dass Verkehr nur über die Gateways der Bastion Hosts stattfindet
  - Bastion Hosts tragen die Anwendungsgateways z.B. auch E-Mail-Proxy mit Virenscanner

Firewall – Filteranordnung



- Demilitarisierte Zone (DMZ) “Niemandsland” enthält Server, die von außen zugänglich sein sollen, z.B.:
  - WWW-Server
  - FTP-Server
- DMZ ≠ Firewall: Separate Firewalls zum Schutz der DMZ und des Innennetzes nötig
- Wenn ein Angreifer einen Server-Host übernehmen konnte, versucht er von dort aus, das Innennetz anzugreifen

Typische Bedrohungen im Internet (Internet Security Threats)

Mapping und Scanning:

- Vor dem eigentlichen Angriff: Erkunde das Netz, finde heraus, welche Hosts, Dienste, Betriebssysteme vorhanden sind
- `ping` kann zeigen, welche Host-Adressen vergeben sind
  (auch Verzeichnisse sind nützlich)
- Port-Scanning: Versuch, zu jedem TCP Port eine Verbindung aufzubauen bzw. jeden UDP-Port anzusprechen
  Kommt eine Reaktion, welche?
  Bekannte Schwachstellen und Angriffsmuster durchspielen.
  - » nmap (http://www.insecure.orig/nmap/) mapper: “network exploration and security auditing”
- Ferner: Versuch, sich einzuloggen, Versuch FTP-Server-Account anzusprechen. Nutzernamen und Passwörter raten.
  Defaultmäßig eingerichete Accounts antesten.

Schutzmaßnahmen?

Internet Security Threats: Schutzmaßnahmen

Verkleinere Angriffsfläche

- Firewalls
- Auf Desktop-PC: Personal Firewall
- Gehärtete Konfiguration

Bemerke Besonderheiten

- Log-Erzeugung und Prüfung (Logging and Audit)
- Verkehrsstatistiken führen und überwachen
- Systemkonfiguration und Dateien überwachen (Tripwire)
- IDS – Automatische Angriffserkennunng (Intrusion Detection Systeme)

Entferne Schwachstellen

- Aktualisiere Systeme, wenn Patches verfügbar
- Scanne selbst, um Schwachstellen zu finden

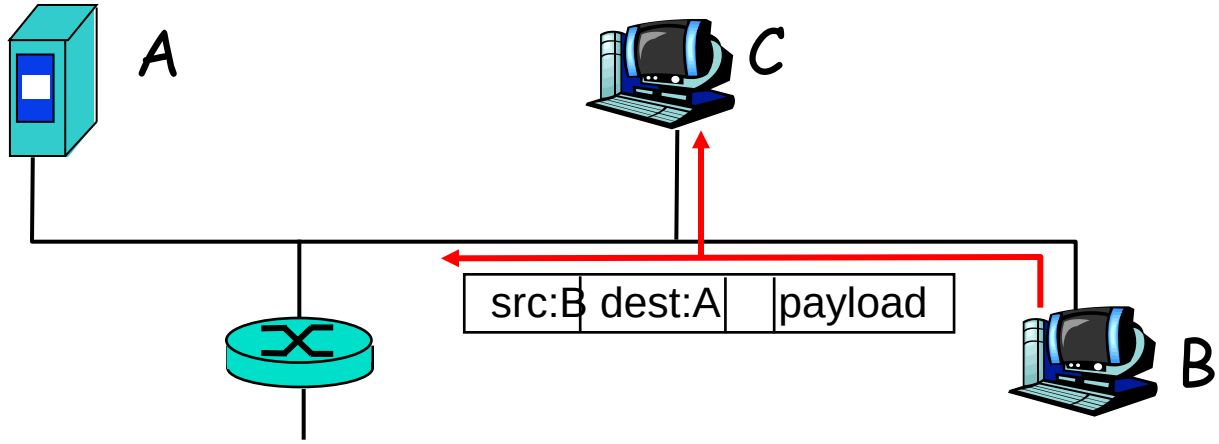
Wehre bösartigen Code ab

- Virenscanner, Firewall, gehärtete Konfiguration, eingeschränkte Nutzeraccounts

Internet Security Threats

Auch das Innennetz ist nicht sicher: Packet Sniffing

- Ethernet hat Broadcast-Segmente
- Angreifer kann seinen NIC so einstellen, dass er jedes Paket mitliest (promiscuous Mode)
- nicht-verschlüsselte Daten können gelesen werden (e.g. Passwörter)
- verschlüsselte Pakete können wieder eingespielt werden
- e.g.: C snifft Bs Pakete



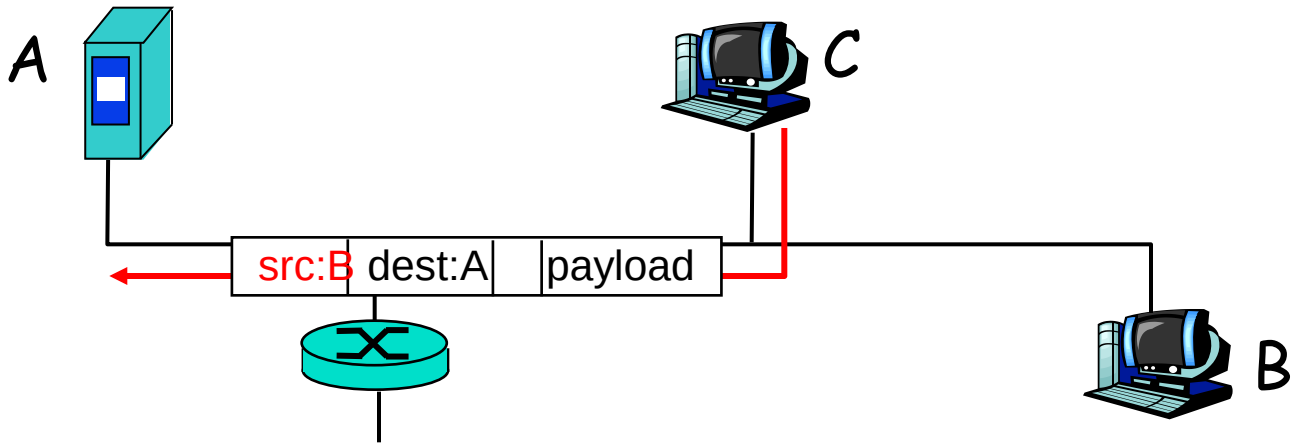
Schutzmaßnahmen?

- 1 Host per Segment (Switches)
- geschützte VPN-Verbindungen

Internet Security Threats

IP-Spoofing:

- Der Sender eines IP-Pakets fälscht die Absender-Adresse
- Der Empfänger kann nie sicher sein, dass die Absender-Adresse stimmt
- e.g.: C pretends to be B



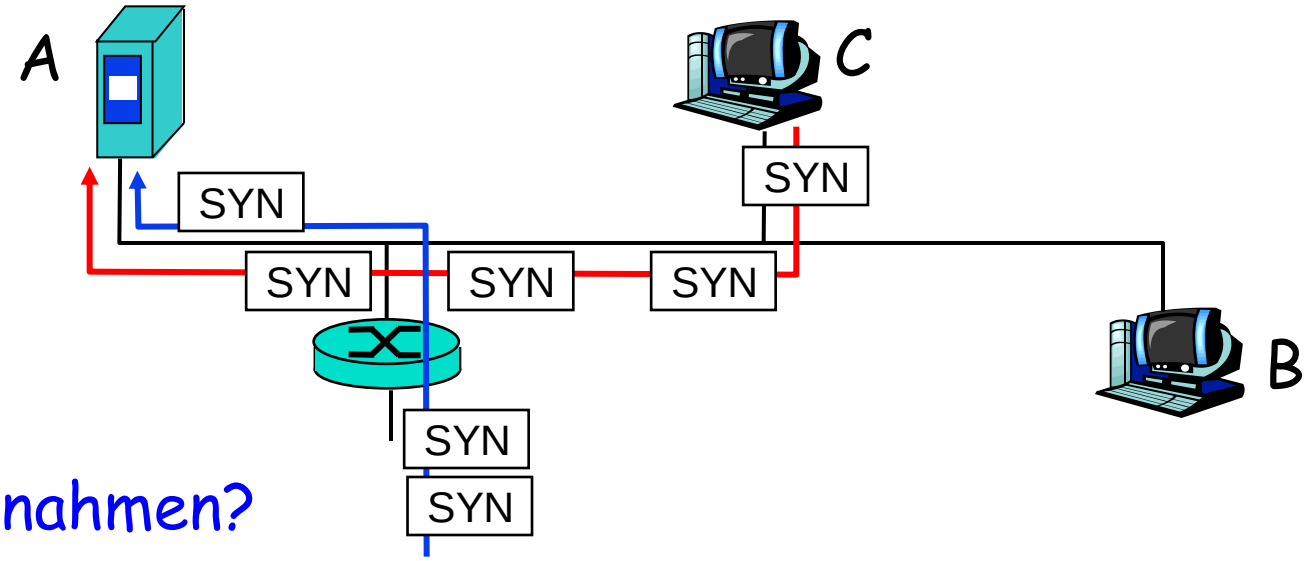
Schutzmaßnahmen?

- Paketfilter enthalten Anti-Spoofing Regel
  (Grober Schutz gegen Adressbereichs-übergreifendes Spoofing)
- authentifizierte VPN-Verbindungen

Internet Security Threats

Verfügbarkeitsangriffe (Denial of Service Attacken – DoS):

- Flut böswillig generierter Pakete überlastet den Empfänger
- Distributed DoS (DDoS): koordinierte Angriffe vieler Sender (z.B. durch von Trojanern verseuchten Internet-User-PCs aus)
- e.g., SYN-Angriff (führt zu halboffenen TCP-Verbindungen)

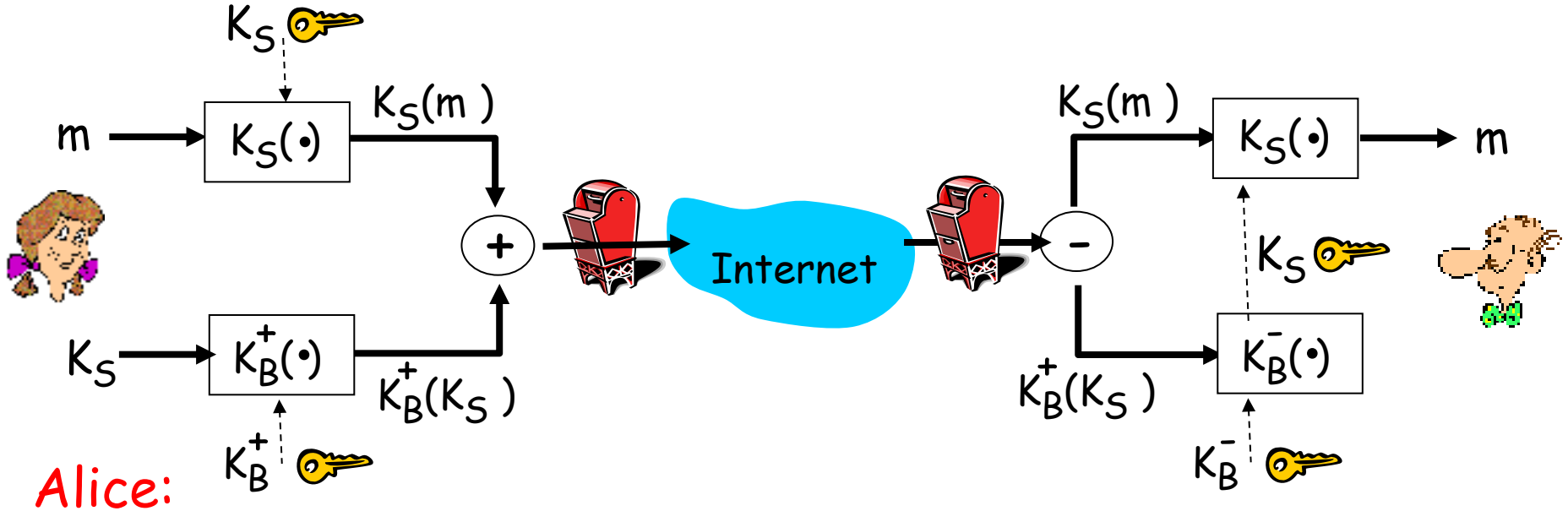


Schutzmaßnahmen?

- Herausfiltern (Firewall) - Problem: Wie trennt man Gute von Schlechten?
- Rückverfolgen

Sichere E-Mail: Vertraulichkeit

- Alice will vertrauliche Mail m an Bob senden
- Bob hat zertifizierten öffentlichen Schlüssel

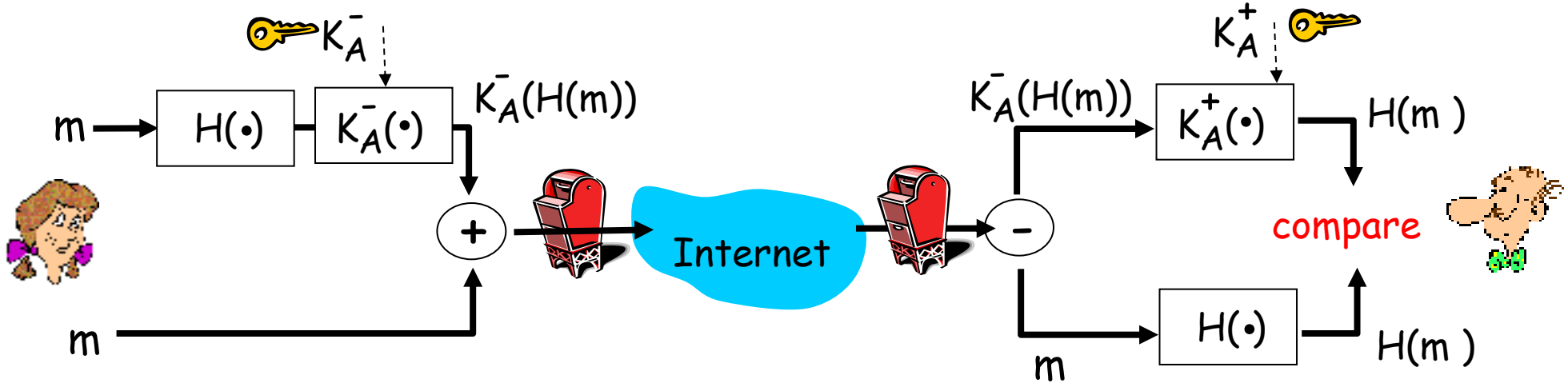


Alice:

- Prüft Bobs Zertifikat: Gültig?
- Generiert per Zufallsgenerator *symmetrischen* Secret Key K_S
- Verschlüsellt Nachricht mit K_S (Effizienz)
- verschlüsselt K_S mit Bobs öffentlichem Schlüssel
- sendet beides, $K_S(m)$ und $K_B(K_S)$, in E-Mail an Bob
- Bob entschlüsselt erst $K_B(K_S)$, dann $K_S(m)$

Sichere E-Mail: Integrität und Authentizität

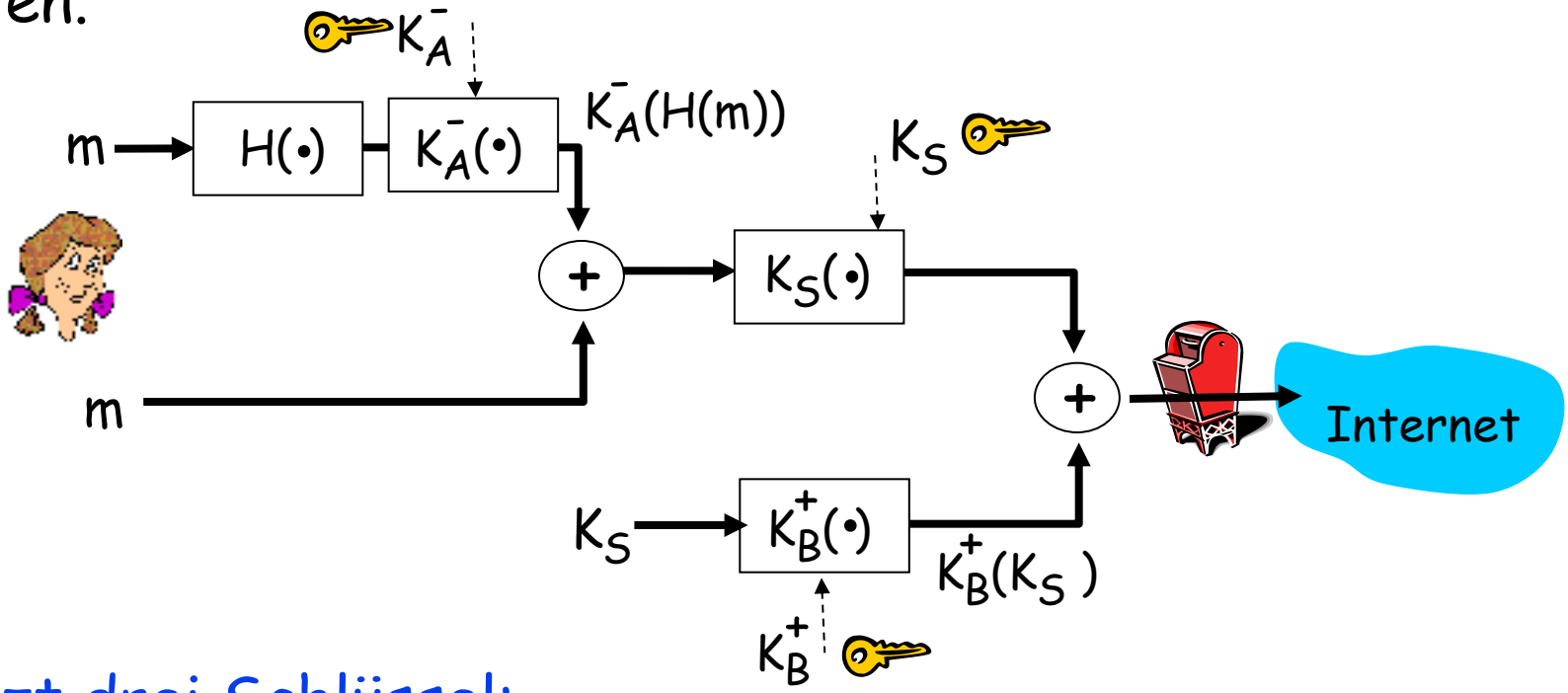
Alice möchte, dass Bob von der Authentizität und Integrität der Mail ausgehen kann



- Alice signiert ihre Nachricht digital
- sie sendet Klartextnachricht, Signatur und Zertifikat

Sichere E-Mail: Vertraulichkeit, Integrität und Authentizität

Alice möchte Vertraulichkeit, Integrität und Authentizität gewährleisten.



Alice benutzt drei Schlüssel:

Ihren eigenen privaten Schlüssel, Bobs öffentlichen Schlüssel und einen zufällig erzeugten symmetrischen Schlüssel

Sichere E-Mail: Problem PKI

PKI: Public Key Infrastructure

1. anerkannte Certification Authorities (CAs)
2. Nutzer müssen dort auch ein Zertifikat haben

Kosten der Zertifikate

Interessant

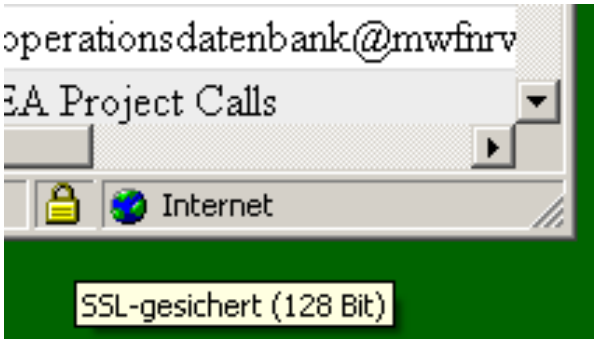
„Billige" Lösungen

z.B. PGP Web of Trust:

Nutzer zertifizieren sich gegenseitig

TLS / SSL: Transport Layer Security / Secure Socket Layer

- "Aufsatz" auf TCP-Verbindungen:
  - (optionale) Authentifikation der Partnerprozesse
  - Vertraulichkeit, Integrität und Authentizität der Nachrichten per Verschlüsselung
- in Anwendungsprozessen zu implementieren, z.B. im Web-Browser und im Web-Server (shttp)
- Betrieb in 2 Phasen
  1. Vorbereitung
     - Authentifikation, Kryptoparameterabstimmung, Sitzungsschlüsselaustausch
  2. Kommunikation "Wie TCP" über Sockets

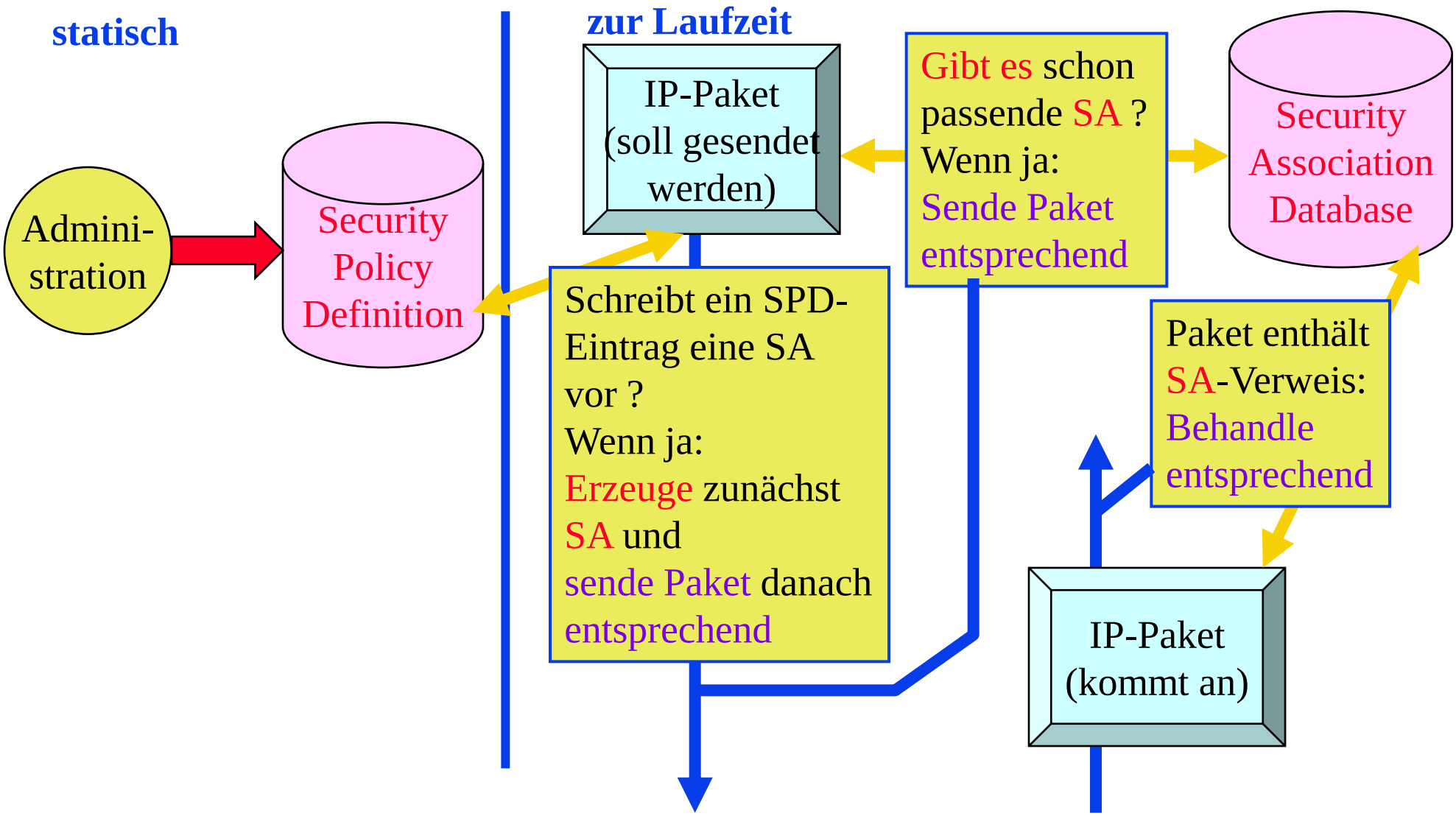


- Server Authentifikation:
  - SSL-Enabled Browser enthält Zertifikate vertrauenswürdiger CAs.
  - Browser fordert von einem kontaktierten Server dessen Zertifikat an, das von einer dieser CAs ausgestellt sein muss
  - Browser prüft mit dem CA-Zertifikat, ob das Server-Zertifikat gültig ist (Problem: Rückrufe)
- Schauen Sie mal in die Einstellungen Ihres Browsers um die CA-Liste einzusehen

IPsec: Network Layer Security

- IPsec ist im Protokoll IP V6 enthalten
  Es kann auch in IP V4 eingesetzt werden
- IPsec sichert den IP-Paketaustausch zwischen Netzknoten
- IPsec wird als “Aufsatz” auf IP im Kern des Host-Betriebssystems implementiert und durch Administrationsparameter aktiviert
  - Vorteil: Keine Änderungen oder Ergänzungen der Anwendungsprozesse nötig
  - Nachteil: Knoten und nicht individuelle Anwendungsprozesse bilden die Endpunkte der gesicherten Kommunikation
- Problem:
  - IP ist verbindungslos/sitzungslos
  - Effiziente Kommunikation verlangt Sitzungsschlüssel als Shared Secret
- Lösung: Konzept der Security Association SA
  - Je Paar aus Quelle und Ziel (also auch je Richtung) wird SA definiert
  - Alle passenden IP-Pakete gehören zur SA, solange SA existiert
- Betrieb ähnlich SSL: 2 Phasen
  - SA Aufbau
  - Paketaustausch
- SA-Aufbau wird durch Knotenadministration gesteuert:
  Security Policy Definition (SPD) legt für “Quelle → Ziel” fest, ob und mit welchen Parametern eine SA einzurichten ist, so dass die IP-Pakete, die diesem Muster folgen, nur innerhalb einer solchen SA ausgetauscht werden.

IPsec: Network Layer Security



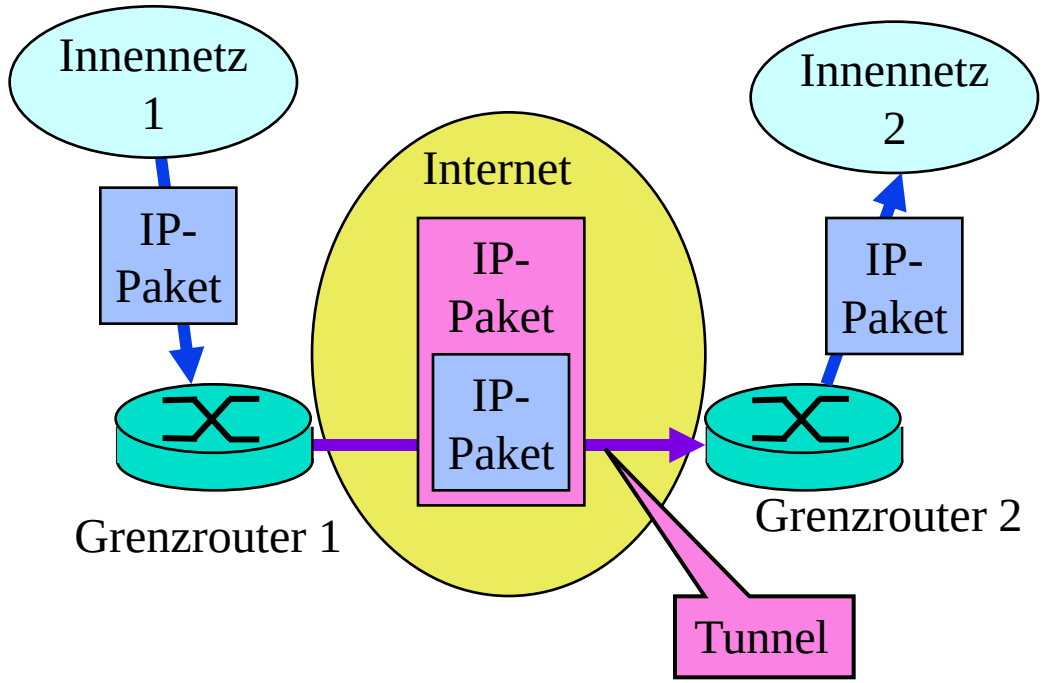
IPsec: Network Layer Security

Es gibt zwei **Betriebsarten**

- Transportmodus
  - gesicherte Kommunikation zwischen Anwendungsprozessen (fast wie SSL, aber nicht durch Anwendungsprozess selbst, sondern durch Knotenadministrator eingerichtet)
- Tunnelmodus
  - gesicherte Kommunikation zwischen Knoten insgesamt
  - Nutzung zur Bildung Virtueller Privater Netze (VPNs)
- **VPN-Bildung**
  - Firmennetz besteht aus Filialnetzen
  - Sie werden über das öffentliche Internet verbunden
  - Die Grenzrouter der Filialnetze richten dazu zueinander IPsec Tunnel ein

TCP/UDP-PDU wird als IP-Paket-Nutzdatum geschützt

IP-Paket wird in neues IP-Paket als Nutzdatum verpackt und geschützt



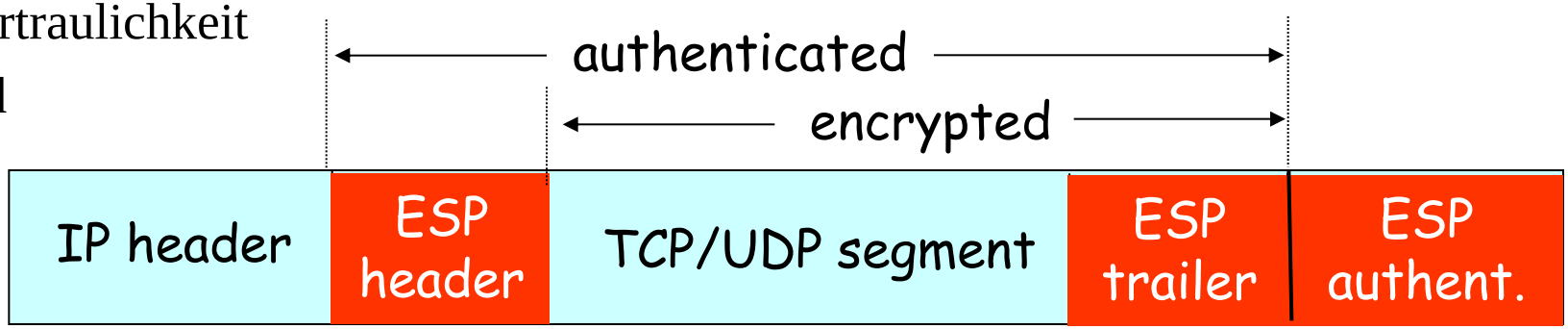
IPsec: Network Layer Security

Es gibt zwei **Schutzfunktionen**, jede entspricht einem Zusatzheader des IP-Pakets

- Authentication Header (**AH**)

  - Authentifikation von Absender und Empfänger
  - Integrität
  - keine Vertraulichkeit

- Encapsulated Security Payload (**ESP**)



  - Authentifikation, Integrität und Vertraulichkeit

Die Art der Schutzfunktion und die Parameter werden durch die **SA** bestimmt
(die ihrerseits wieder durch einen SPD-Eintrag bestimmt wird)

IEEE 802.11 Wireless LAN – Security

- *WLAN-Frames können leicht abgehört werden*
  - *Funkwellen halten sich nicht an die Grundstücksgrenzen*
  - *es gibt Richtantennen*
- Sicherheitsfunktionen
  - Authentifikation und Verschlüsselung

- Wired Equivalent Privacy (WEP): Ein schwacher Versuch
  - Authentifikation a la *ap4.0, Shared Secret* und *Challenge Response* basiert
    - » Host sendet Request an Access Point, der antwortet mit 128-Bit Nonce
    - » Host sendet verschlüsselte Nonce zurück
  - Keine dynamische Schlüsselverteilung
  - Es gibt für Access Point und alle Hosts ein Gruppen-“Shared Secret“ Daraus werden alle benötigten Schlüssel abgeleitet.
  - Verschlüsselung ist relativ leicht zu brechen

WEP Verschlüsselung

- Host/AP share 40 bit symmetric key
- Host appends 24-bit initialization vector (IV) to create 64-bit key
- 64 bit key used to generate stream of keys, k_i^{IV}
- k_i^{IV} used to encrypt ith byte, d_i, in frame: $c_i = d_i$ XOR k_i^{IV}
- IV and encrypted bytes, c_i sent in frame

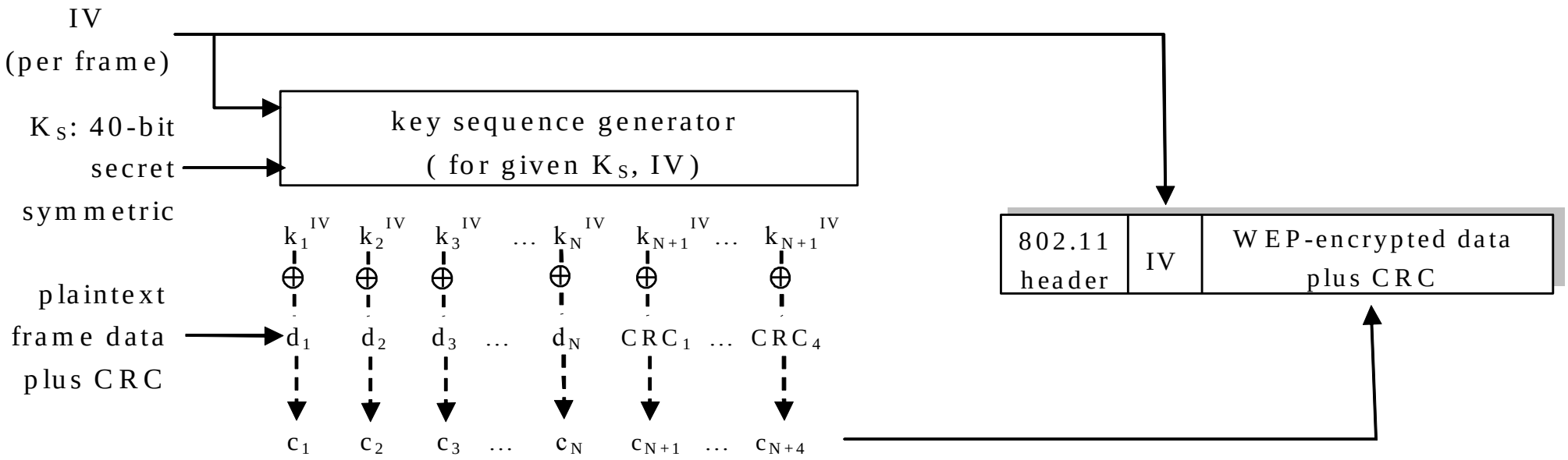


Figure 7.8-new1: 802.11 WEP protocol

Brechen der 802.11 WEP Verschlüsselung

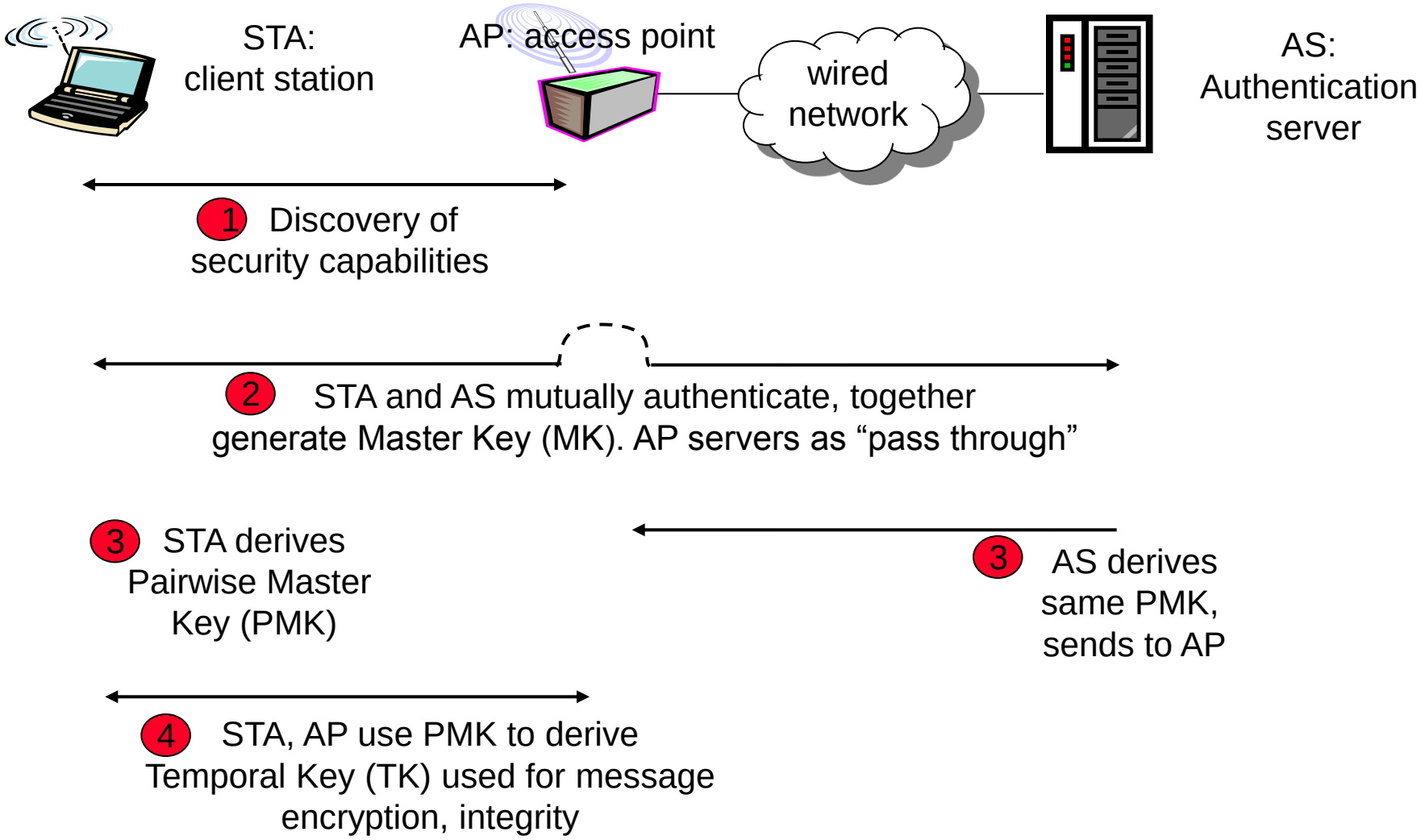
Schwachstelle:

- 24-bit IV, one IV per frame, → IV's eventually reused
- IV transmitted in plaintext → IV reuse detected
- **Angriff:**
  - Trudy causes Alice to encrypt known plaintext $d_1\ d_2\ d_3\ d_4\ \ldots$
  - Trudy sees: $c_i = d_i$ XOR k_i^{IV}
  - Trudy knows $c_i\ d_i$, so can compute k_i^{IV}
  - Trudy knows encrypting key sequence $k_1^{IV}\ k_2^{IV}\ k_3^{IV}\ \ldots$
  - Next time IV is used, Trudy can decrypt!

802.11i: Verbesserte Sicherheit im WLAN

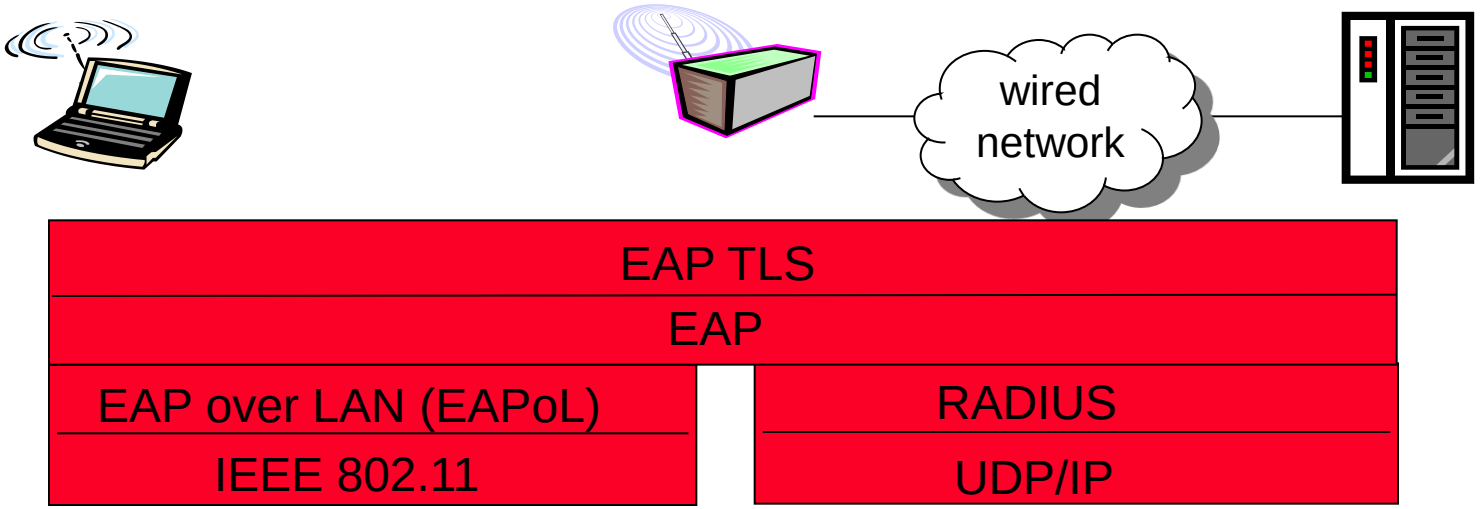
- man kann deutlich stärker verschlüsseln
- dynamische Schlüsselverteilung wird unterstützt
- bindet einen separaten Authentifikationsserver ein, der nicht mit dem Access Point zusammenfällt
  (z.B. Kerberos, RADIUS)

802.11i: Vier Phasen des Betriebs



EAP: Extensible Authentication Protocol

- EAP: Protokoll zwischen mobilem Client und dem Authentifikationsserver
- Ist erweiterbar, d.h. kann verschiedene Authentifikationsverfahren einbetten, z.B. RADIUS
- Authentifikation über verschiedene Teilstrecken abgewickelt
  - mobiler Client – Access Point (EAP over LAN)
  - Access Point – Authentifikationsserver (RADIUS over UDP)



Kap. 7: Sicherheit im Netz

Lernziele:

- Prinzipien der Sicherheit im Netz:
  - Kryptographie und Nutzungen, die über Vertraulichkeitsschutz hinausgehen
  - Authentifikation
  - Nachrichtenintegrität
  - Schlüsselverteilung
- Sicherheit in der Praxis:
  - Firewalls
  - Sicherheitsfunktionen in den Kommunikationsschichten