openHPI-Kurs: Wie funktioniert das Internet?

Die Technik des Internetworking

Prof. Dr. Christoph Meinel
Hasso-Plattner-Institut
Universität Potsdam

Einführung (1/2)

Suche nach dem einen Netz…

- Haben inzwischen sehr verschiedene Netzwerktypen und Technologien kennengelernt
- Die einzelnen Netztechnologien unterscheiden sich grundlegend und sind untereinander vollkommen unverträglich
- Jede Netztechnologie besitzt ihre
  - spezifischen Medien,
  - Adress-Schemata und
  - Paketformate
- Jede Netztechnologie hat ihre Vor- und Nachteile und ist jeweils für einen besonderen Einsatz geeignet, keine erfüllt alle Bedürfnisse

Einführung (2/2)

Wie lassen sich unterschiedliche Netze verknüpfen?

Herausforderungen:

- Wie wird zwischen den verschiedenen Netztechnologien vermittelt?
- Einheitliches Paketformat und Adress-Schema …
- Wie finden Datenpakete in einem komplexen heterogenen Netzwerkverbund ihr Ziel?
- Wie werden die Paketrouten berechnet? Wie kommt der Paketvermittler (Router) zu seiner Routingtabelle?
- Wie werden Staus und Überlastsituationen vermieden?
- Wie werden Übertragungsfehler vermieden/überwunden?

Internetworking

Internetworking – Wiederholung (1/2)

Internetworking

- Konzept zum Zusammenschluss **heterogener** Computernetze zu einem einheitlichen Kommunikationssystem

Internet

- Zusammenschluss mehrerer heterogener physikalischer Netze mit unterschiedlichen Netzwerktechnologien zu einem Netzwerk
  → **virtuelles Netz**

Wichtigstes Beispiel

- Das weltweite Internet

Internetworking – Wiederholung (2/2)

Grundidee

- Zusammenschluss der verschiedenen Netztechnologien erfolgt mit Hilfe von Spezialrechnern → „**Router**“
- Einführung eines einheitlichen Protokollstapels oberhalb der technologiegebundenen Ebene lässt Netzwerkverbund als logisch nahtloses Kommunikationssystem erscheinen

  → **virtuelles Netz**
- Alle angeschlossenen Hosts und Router müssen über diese Internet-Protokollsoftware verfügen
- Am häufigsten eingesetzte Internet-Protokollsoftware:
  - **TCP/IP-Protokollsuite** bzw. **TCP/IP-Stapel** (TCP/IP-Stack)

Zwischen- und Vermittlungssysteme

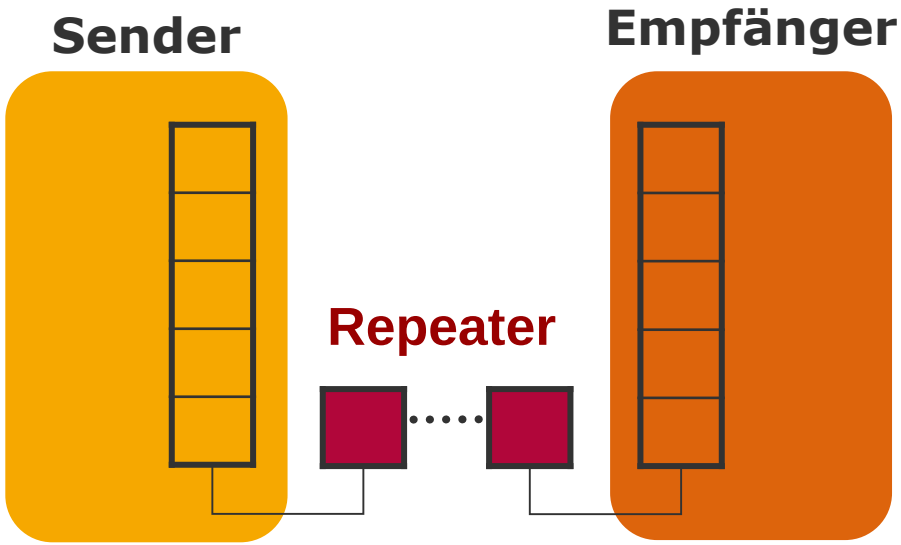
Was hält das Internet zusammen?

- Die am Internet beteiligten physikalischen Netze werden mit Hilfe von Zwischensystemen verknüpft, die die Aufgaben der Paketvermittlung zwischen den Netzen übernehmen
- Es gibt Zwischensysteme mit verschiedenen Funktionsumfängen:
  - Repeater
  - Hub
  - Bridge (Switch)
  - Router
  - Gateway
- Funktionsumfang legt Einordnung in bestimmte Protokollschicht des TCP/IP-Protokollstapels fest

Repeater

- Arbeiten auf unterster Schicht (physikalische Schicht), sind daher für höhere Schichten „unsichtbar“
- Dienen der reinen Signalverstärkung bei der Überwindung größerer Distanzen
- Verfügen über keinerlei „Eigenintelligenz“

4	Anwendung
3	TCP (Transport)
2	IP (Internet)
1	Netzzugang (LAN)
0	Hardware



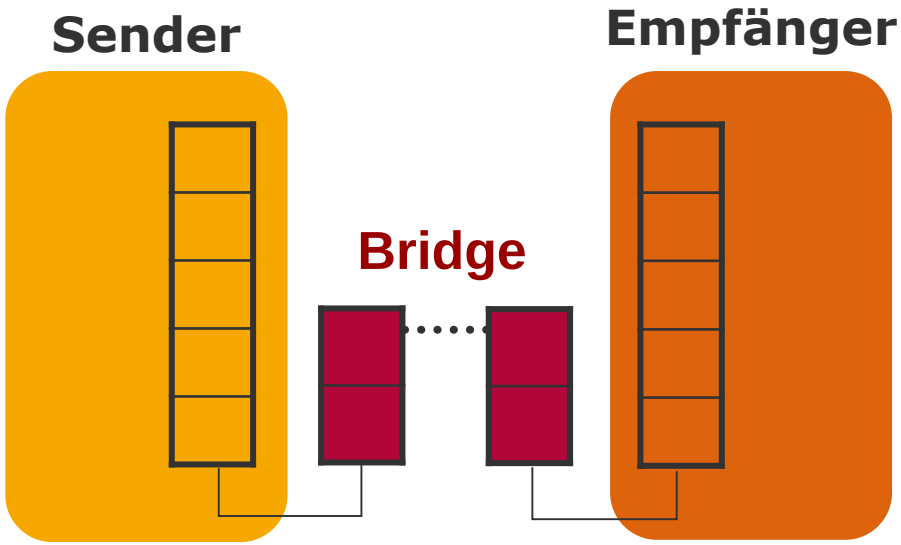
Bridge und Switch

Bridge (verbindet zwei LAN-Segmente)

Switch (verbindet mehrere LAN-Segmente)

- Ermöglichen LAN-Erweiterung mit **intelligentem Verkehrsmanagement**
- Lokaler Verkehr bleibt lokal, wird nicht über Bridge weitergeleitet

4	Anwendung
3	TCP (Transport)
2	IP (Internet)
1	Netzzugang (LAN)
0	Hardware



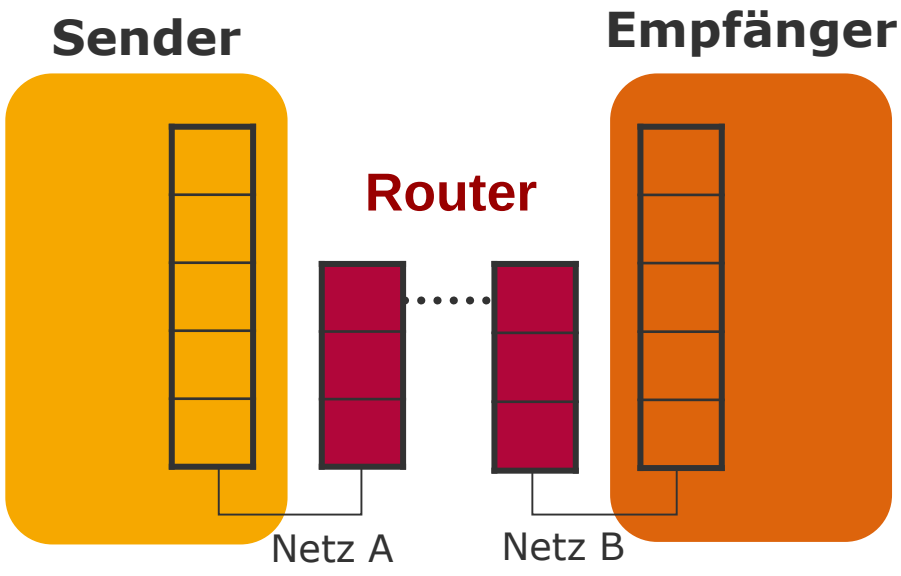
Router (1/3)

- Spezialrechner, die physikalische Verbindung zwischen verschiedenen Netzen herstellen
- Bestehen aus Prozessor, Speicher und E/A-Schnittstelle (Netzwerkkarte) für jedes angeschlossene Netz
- Einzige Aufgabe ist die Beförderung von Datenpaketen zwischen den angeschlossenen Netzwerken
- Setzen Adressschemata, Paketformate usw. der einen Netzwerktechnologie in die des anderen Netzwerks um
- Router müssen dazu wie „normale“ Endgeräte gleichzeitig zu allen angeschlossenen Netzwerken gehören

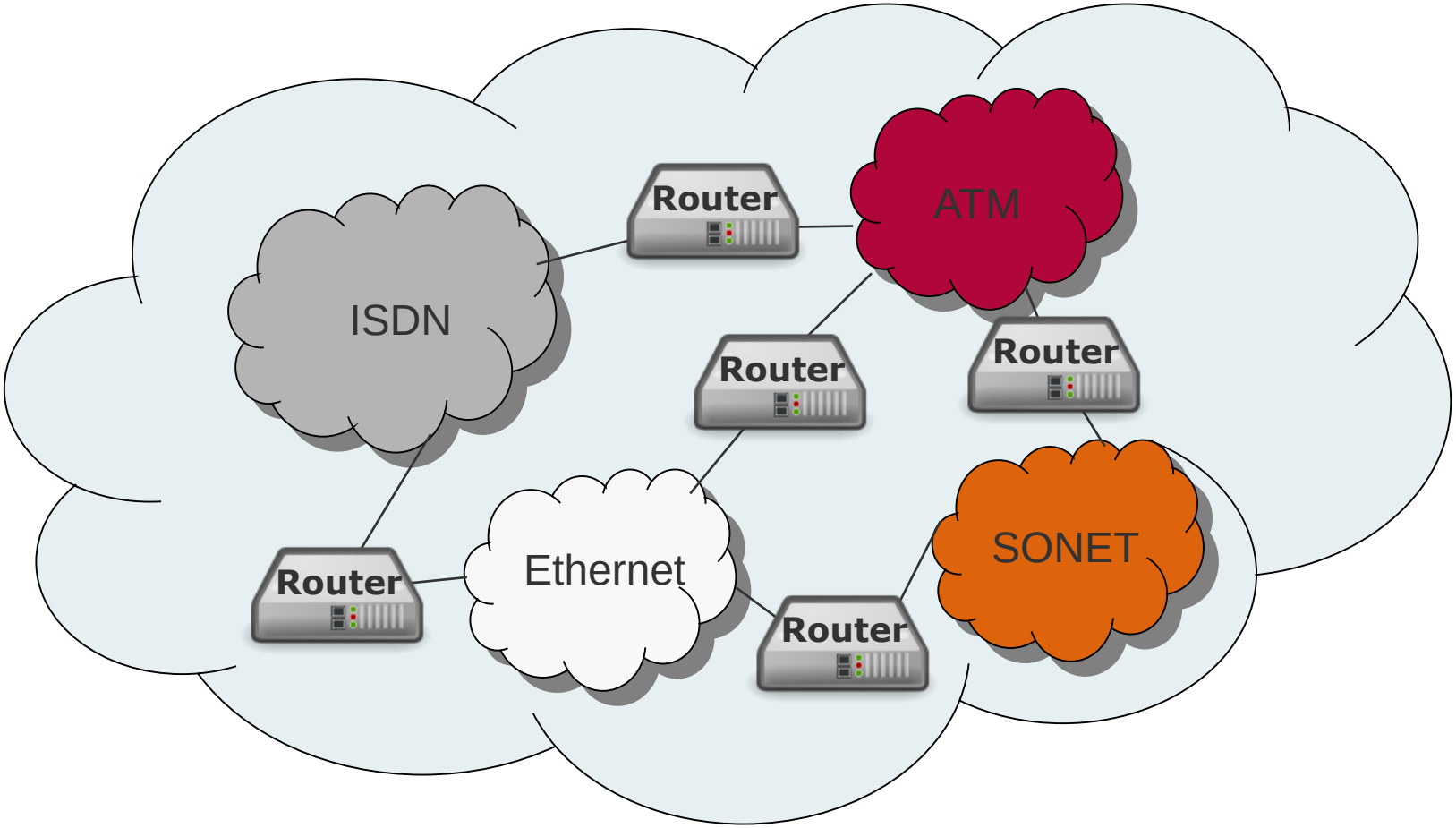
Router (2/3)

- Verbindet zwei autarke Netze zu einem Internet
- Subnetze werden logisch auf Schicht 2 getrennt
- Netzwerktopologie und -Technologie muss dem Router bekannt sein, um Datenpakete effizient weiterleiten zu können

4	Anwendung
3	TCP (Transport)
2	IP (Internet)
1	Netzzugang (LAN)
0	Hardware



Router (3/3)

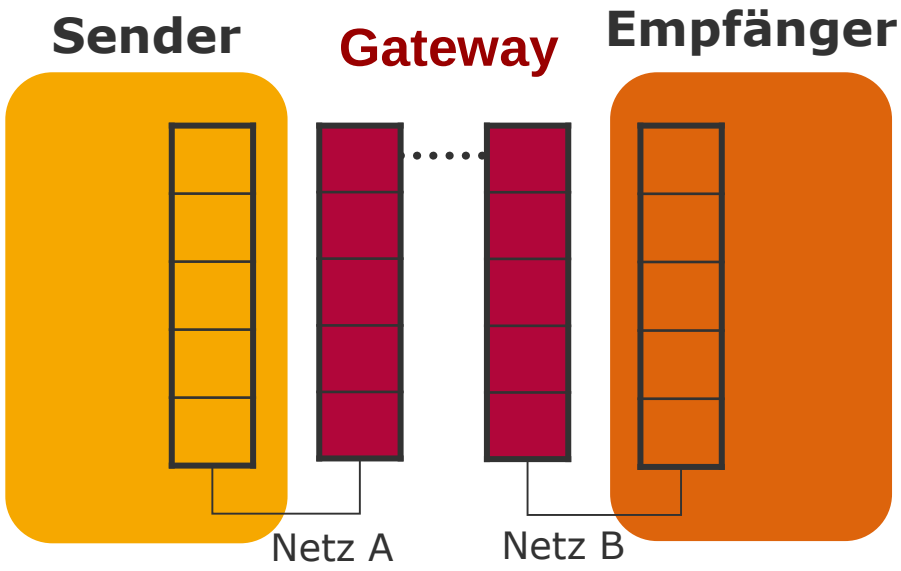

Router
ATM
ISDN
Router
Router
Router
Ethernet
SONET
Router
Router

Gateway

Gateway (genauer: Application Level Gateway)

- Verbindet Netzwerke auf der Anwendungsebene
- Dient zur Kommunikation zwischen Anwendungen bzw. Komponenten einer verteilten Anwendung auf unterschiedlichen Endsystemen
- Übersetzt unterschiedliche Anwendungsprotokolle ineinander

4	Anwendung
3	TCP (Transport)
2	IP (Internet)
1	Netzzugang (LAN)
0	Hardware



openHPI-Kurs: Wie funktioniert das Internet?

Internetprotokoll IP

Prof. Dr. Christoph Meinel
Hasso-Plattner-Institut
Universität Potsdam

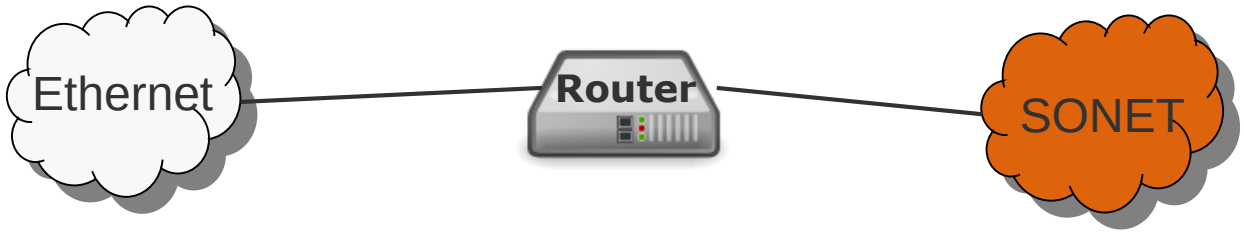
Internetprotokoll IP

Internetprotokoll ist zentrales Netzwerkprotokoll im TCP/IP-Stack auf der **Internetschicht / Internet Layer**

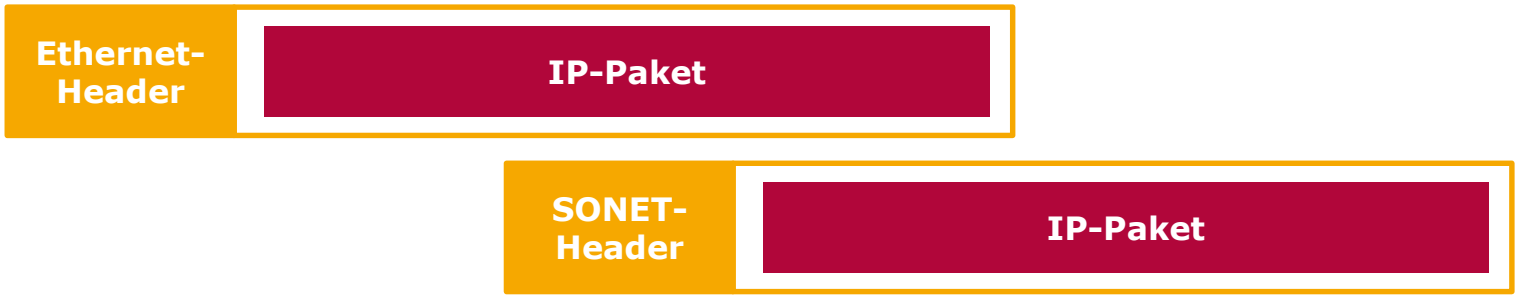
- Aufgaben:
  - Vermittlung zwischen unterschiedlichsten Netztechnologien
  - Bereitstellung eines einheitlichen Adressierungsschema im weltweiten Internet
  - Korrekte Weiterleitung von Internetdatenpaketen über den Netzwerkverbund trotz unterschiedlicher Betriebsparameter
  - Auffinden optimaler Wege für Datenpakete (Paketrouten)
  - Übertragungsfehler erkennen anhand von Prüfsummen
- Aktuell verwendete IP-Versionen:
  - IPv4
  - IPv6

Aufgaben des Internetprotokolls (1/4)

- Unterschiedliche Netzwerke sind im Internet durch **Router** verbunden, z.B.



- IP-Pakete werden in den zu überquerenden Netzwerken als Nutzdaten in den dort gültigen Paketformaten transportiert



- IP-Paket hat keine Kenntnis von Protokollen des Link Layers

Aufgaben des Internetprotokolls (2/4)

Internetprotokoll bietet einheitliches, netzwerkübergreifendes

Adressierungsschema

- Sorgt für Adressumsetzung an Netzwerkschnittstellen
- Benötigt Verzeichnisdienste, um die Zuordnung der unterschiedlichen IP-Adressen zu ermöglichen
  - Weltweit eindeutige IPs, z.B. an Routern)
  - Lokale IPs, die weltweit mehrfach vorkommen können (z.B. im Heimnetzwerk)

Beispiel:

141.89.225.101

IPv4-Adresse

2001:638:807:204::8d59:e17e

IPv6-Adresse

Aufgaben des Internetprotokolls (3/4)

Korrekte Weiterleitung von Datenpaketen bei unterschiedlichen technischen Betriebsparametern der beteiligten Netzwerke

Problem: Maximale Paketgröße

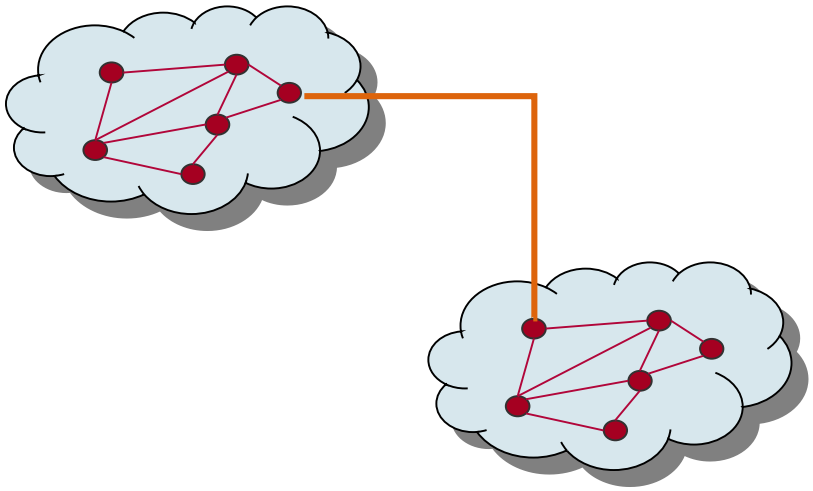
- Trifft Datenpaket auf ein Netzwerk, dessen MTU (Maximum Transfer Unit, maximale Datenpaketgröße) kleiner ist als das weiterzuleitende Datenpaket, muss das Datenpaket entsprechend zerlegt – **fragmentiert** – werden
- Dabei muss jedem Fragment Zusatzinformation beigefügt werden, sodass Empfänger das Datenpaket korrekt wieder zusammensetzen – defragmentieren – kann

Aufgaben des Internetprotokolls (4/4)

Internetprotokoll sorgt dafür, für ein IP-Paket einen möglichst optimalen Weg durch das Internet vom Sender zum Empfänger zu finden

- Internet hat Millionen von Netzwerkknoten
  - Paket soll nicht über unnötig viele Knoten verschickt werden
  - Paket soll möglichst schnell Empfänger erreichen

→ **IP-Routing**



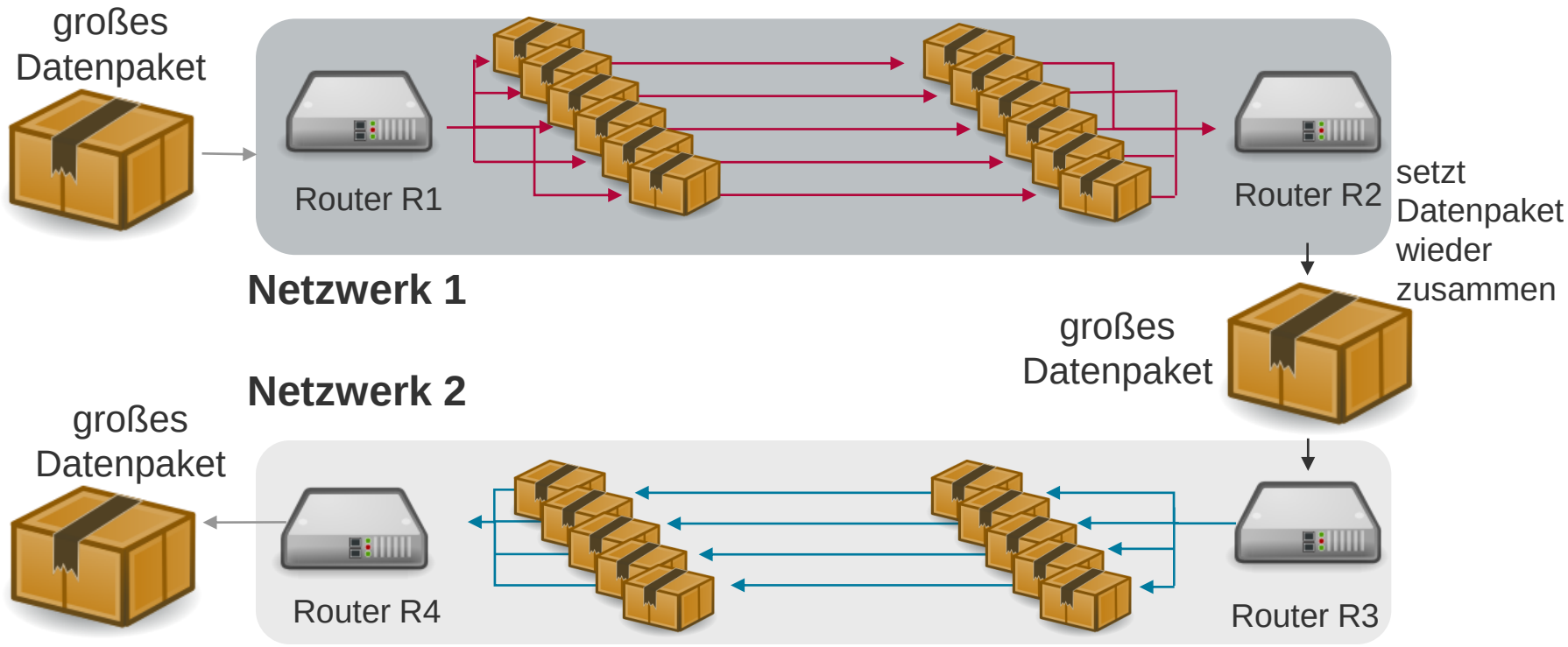
openHPI-Kurs: Wie funktioniert das Internet?

Fragmentierung von Datenpaketen

Prof. Dr. Christoph Meinel
Hasso-Plattner-Institut
Universität Potsdam

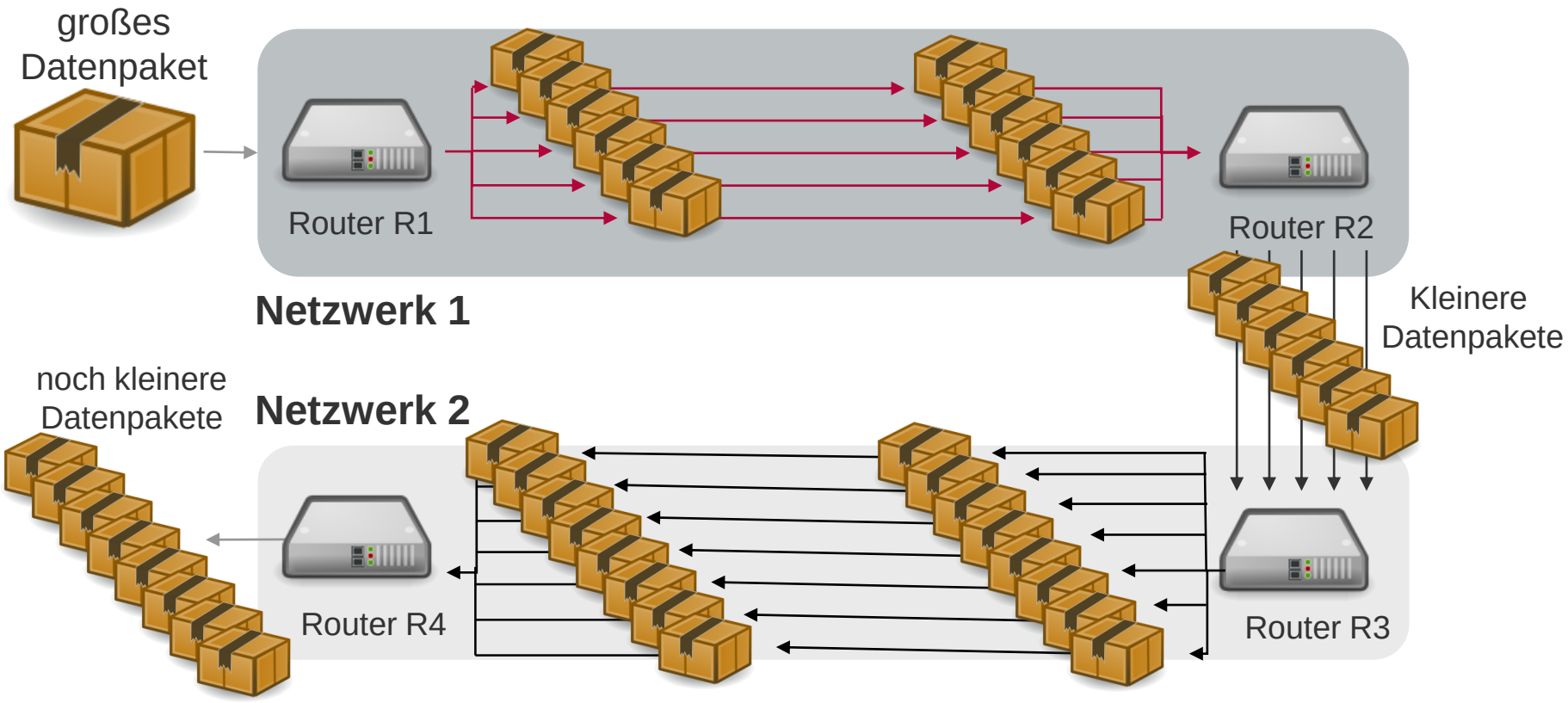
Fragmentierung (1/5)

Transparente Fragmentierung



Fragmentierung (2/5)

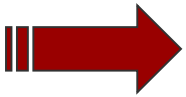
Nicht-transparente Fragmentierung



Fragmentierung (3/5)

Probleme:

- Beim Empfänger müssen die einzelnen Datenpakete wieder **korrekt zusammengesetzt** werden können
- Datenpakete können ihr Ziel auf unterschiedlichen Wegen erreichen
  - es kann zu unterschiedlichen Fragmentierungen kommen
  - **Übertragungsfehler** können **Neuübertragung** eines Fragments erfordern, die entlang eines anderen Weg realisiert wird und dort erneut zu anderer Fragmentierung führt



Wie kann korrektes Zusammensetzen gewährleistet werden?

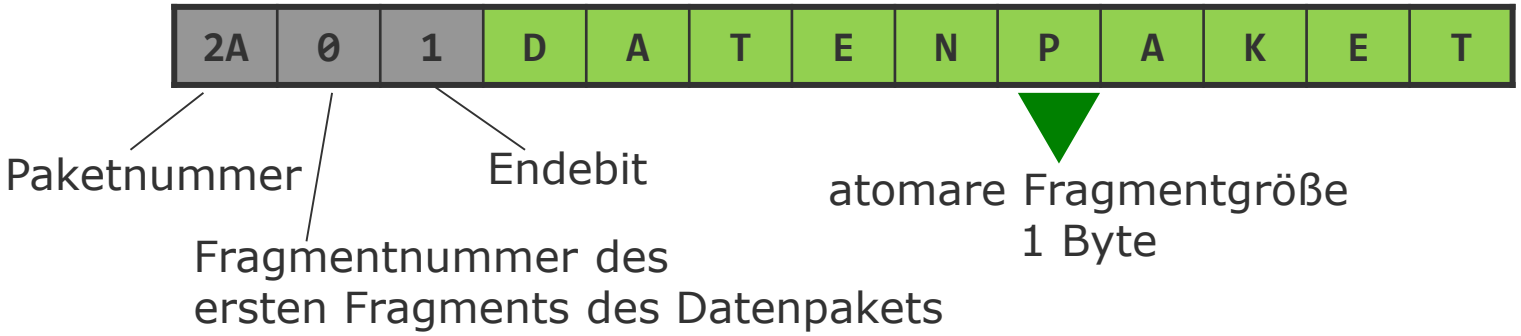
Atomare Fragmentgröße

- kleiner dürfen Fragmente nicht unterteilt werden

Fragmentierung (4/5)

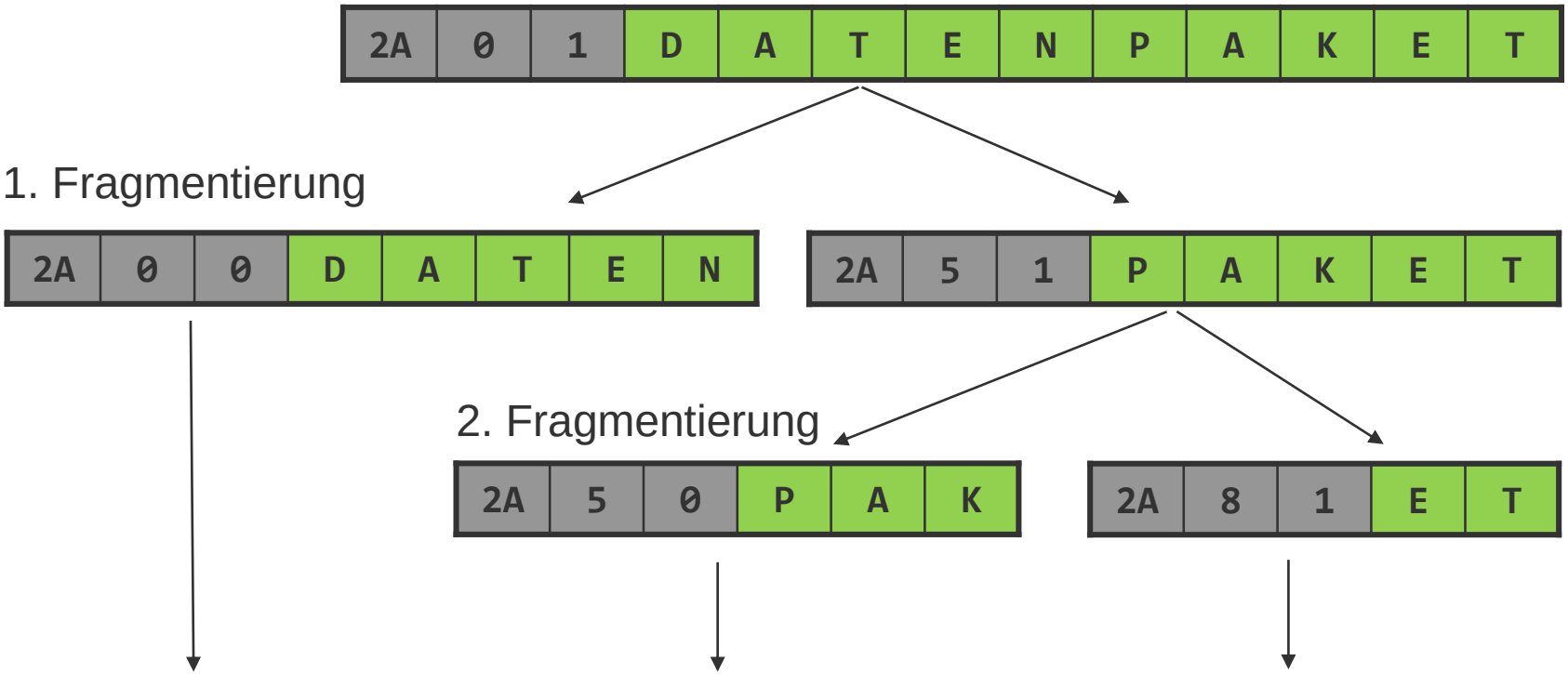
Zusätzliche Informationen im Header des Datenpakets:

- Paketnummer
- Nummer des ersten im IP-Datenpaket enthaltenen Fragments
- Steuerungsbit, das angibt, ob es sich um letztes Fragment eines Datenpakets handelt (=1) oder ob noch weitere Fragmente folgen (=0)



Fragmentierung (5/5)

Beispiel:



openHPI-Kurs: Wie funktioniert das Internet?

Adressierung mit IPv4

Prof. Dr. Christoph Meinel
Hasso-Plattner-Institut
Universität Potsdam

Adressierung mit IPv4 (1/2)

Problem: Wie findet ein Datenpaket seinen Weg zum Ziel?

Einheitliche Adressierung notwendig:

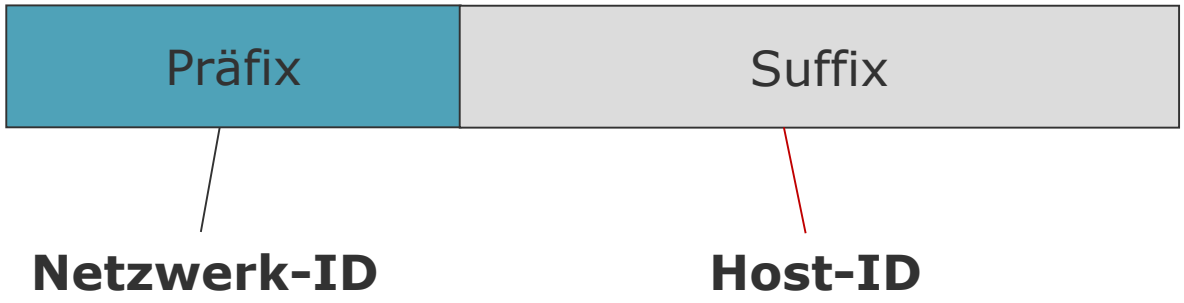
- IP-Protokollsoftware weist dabei jedem Host **eindeutige Adresse** zu
- IP-Standard (**IPv4**) ordnet jedem Host für die gesamte Kommunikation im Internet eine 32 Bit Binärzahl zu:
  ➔ **IP-Adresse** (*Internet Protocol Address*)
- Jedes im Internet versendete Datenpaket enthält die IP-Adresse des Senders und die des Empfängers

Adressierung mit IPv4 (2/2)

IPv4-Adresse (32 Bit) besteht aus

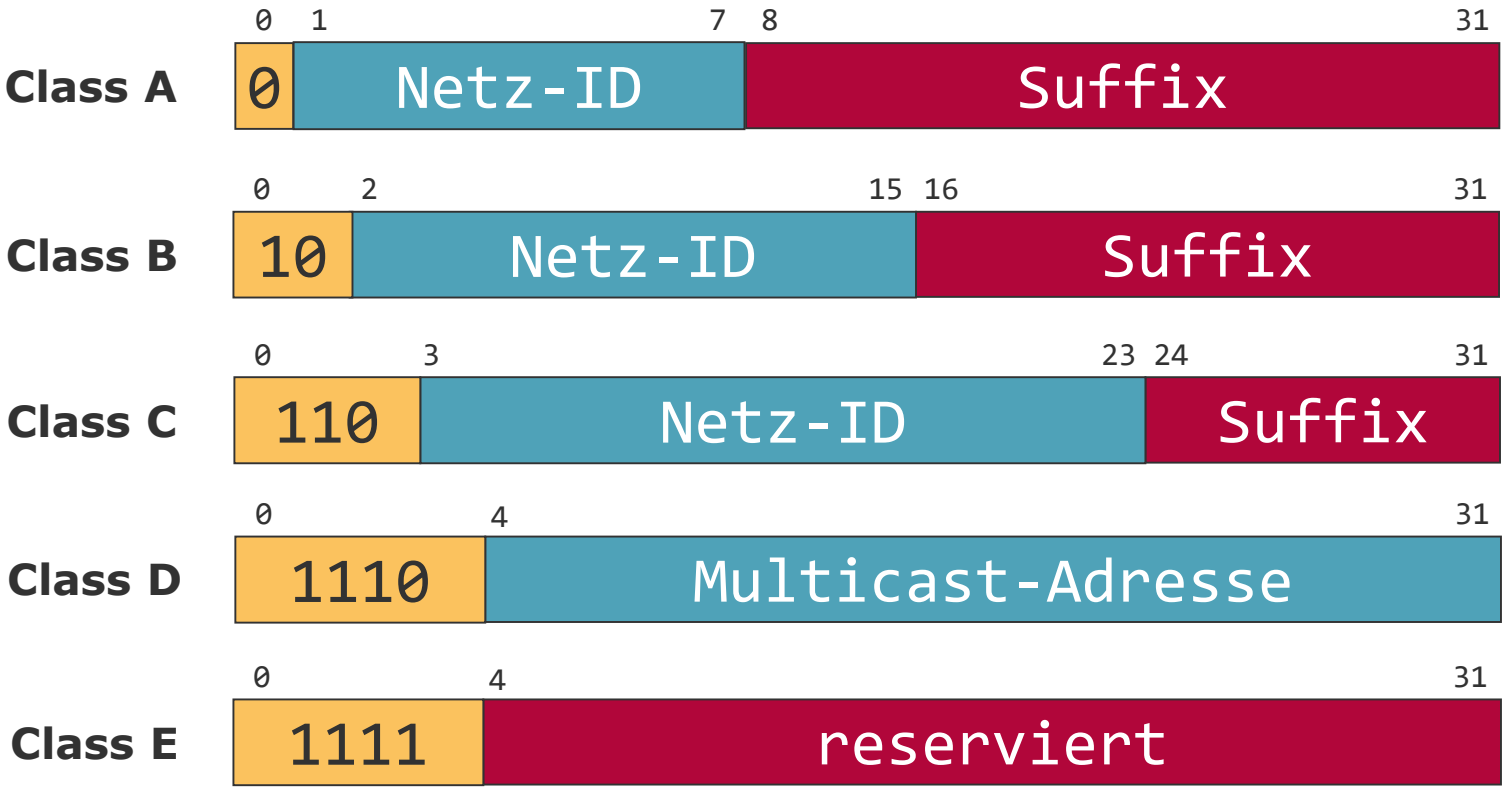
- **Präfix** – identifiziert das physische Netz
- **Suffix** – identifiziert einen bestimmten Computer im betreffenden Netz

IP-Adresse identifiziert also keinen bestimmten Rechner, sondern eine **Verknüpfung von einem Computer mit einem Netz**



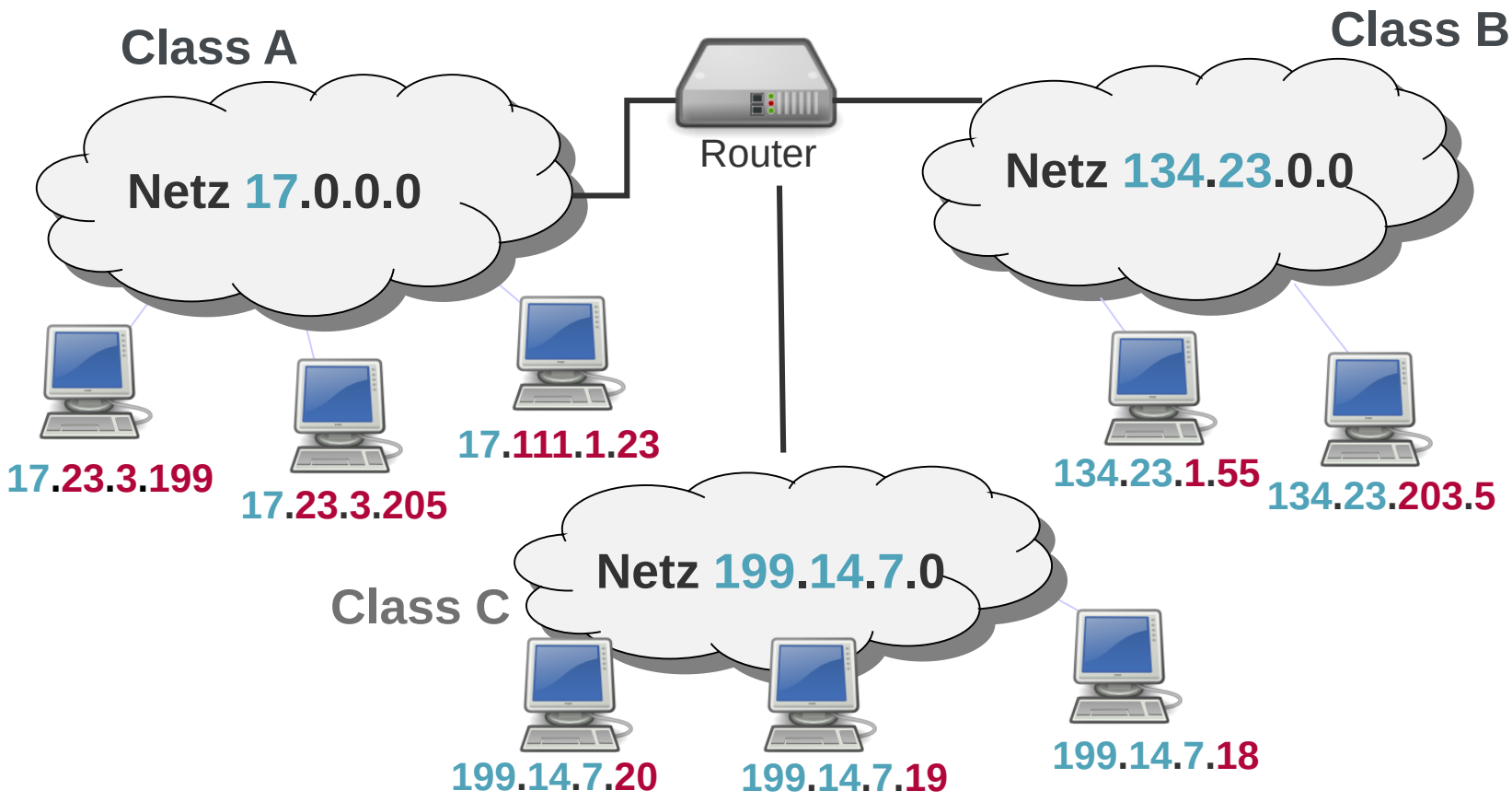
Classful Addressing mit IPv4 (1/4)

Classful Addressing (historisch)



Classful Addressing mit IPv4 (2/4)

Beispiel für Classful Addressing:



Classful Addressing mit IPv4 (3/4)

Spezielle IPv4-Adressen

Broadcast-Adressen

- Host-ID = 255
- All-Network-Broadcast: 255.255.255.255

Netzwerkadressen/Local Net

- Host-ID = 0
- Local Network: Netzwerk-ID = 0

Loop-Back-Adressen (Local Host)

- Netzwerk-ID = 127
- Host-ID = 1 (Konvention)

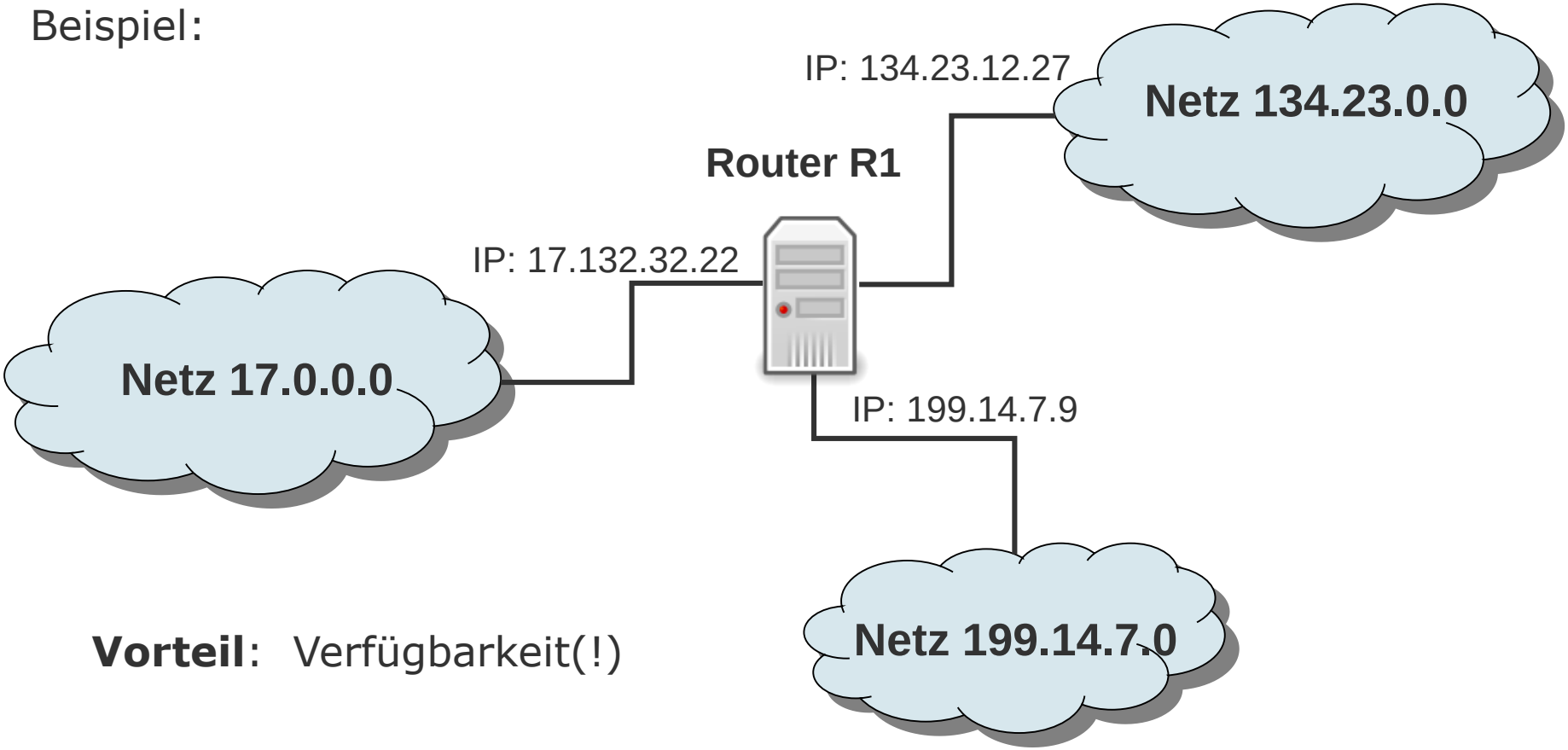
Private Adressen (RFC 1597, ersetzt durch RFC 1918)

- 10.x.x.x, 172.16.0.0 – 172.31.254.254, 192.168.0.0 – 192.168.254.254

Classful Addressing mit IPv4 (4/4)

Multihomed Hosts

Beispiel:



Vorteil: Verfügbarkeit(!)

Adressraum – Theorie und Praxis …

Problem:

- Was tun, wenn das Netzwerk über die maximal mögliche Anzahl von Rechnern wächst?
  - zusätzliche Netzwerk-ID?
  - Netz der nächsthöheren Adressklasse beantragen?
    → Class C Netz (254 Hosts) meist zu klein
    → Class B Netz (65.534 Hosts) meist zu groß

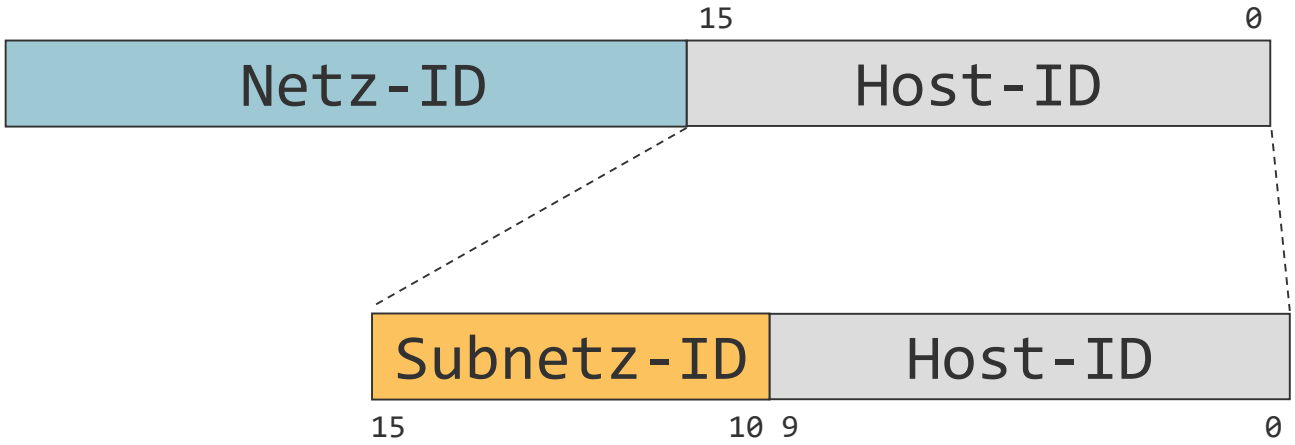
Lösungen:

- Subnetting (Subnetzadressierung)
- Classless Addressing (Supernetting)
  und Classless Inter-Domain Routing (CIDR)
- Network Address Translation (NAT)
  und Network Address Port Translation (NAPT)

Subnetting und Classless Addressing (1/4)

Subnetting: Logische Unterteilung eines Netzwerks in separate, physisch voneinander unabhängige Subnetze

- Ursprünglich: Aufteilung von „classful“ adressierten Netzen in mehrere kleinere Netze



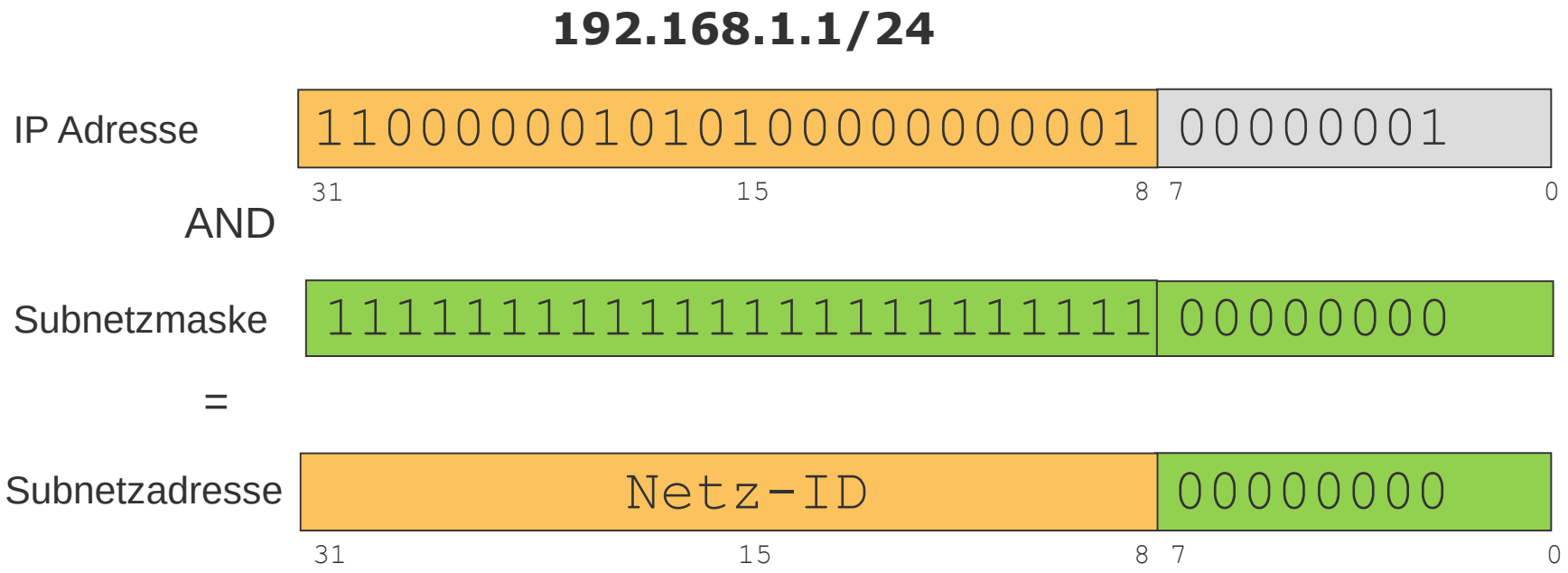
Subnetz-ID mit 6 Bit → 62 Subnetze

Host-ID mit 10 Bit → 1022 Hosts

Wie unterscheidet man Subnetz-ID ←→ Host-ID?

Subnetting und Classless Addressing (2/4)

Heute: Wegfall der Adressklassen und der festen Unterscheidung von Netz- und Host-ID. Stattdessen Einsatz von **Subnetzmasken** variabler Länge (VLSM, Variable Length Subnet Mask)



Subnetting und Classless Addressing (3/4)

Router identifiziert das Netz (Subnet) über zugehörige **Subnetzmaske:**

- **IP-Adresse AND Subnetzmaske = (Sub)Netzwerk-ID**
- **IP-Adresse AND Subnetzmaske^{-1} = Host-ID**
  (Subnetzmaske^{-1} := bitweise Invertierung der Subnetzmaske)

Schreibweise für Netzmasken

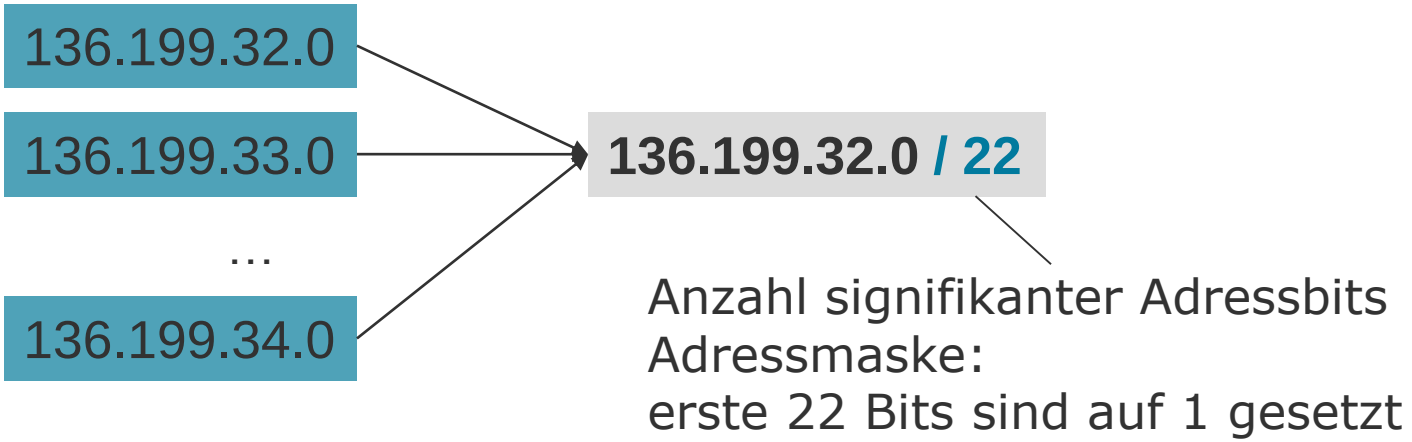
- **Dotted-decimal Notation:**
  192.168.1.0/255.255.255.0
- **CIDR Notation:**
  192.168.1.0/24
  → /24 gibt die Anzahl an 1-Bits in der Subnetzmaske an
  (ab most significant Bit)

Subnetting und Classless Addressing (4/4)

Supernetting ist ein komplementärer Ansatz zu Subnetting

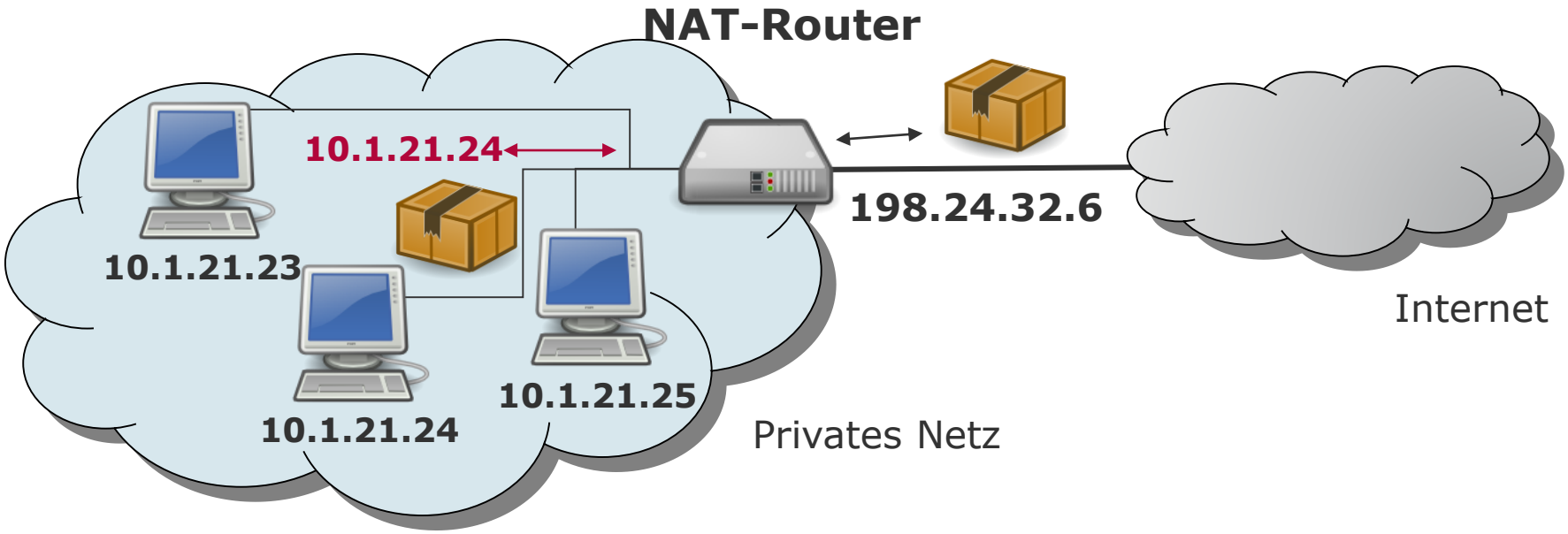
- Einzelne Netzwerke werden zu einem „Supernetz" zusammengefasst, um Zahl der Einträge in Routingtabellen zu reduzieren
- Nutzung zusammenhängender Adressblöcke (CIDR Blocks)
- Classless Interdomain Routing (RFC 1519):

  Eintrag (139.199.32.0,3) spezifiziert die drei Netzwerkklassen



NAT – Network Address Translation

- Nutzung der **„privaten" IP-Adressbereiche**
- Private IP-Adressen dürfen nur **intern** verwendet werden
- **NAT**-Router übersetzt interne/externe IP-Adressen
- **NAPT** (Network Address Port Translation) verwendet zur eindeutigen Identifikation **TCP-Quell-/Zielport** → nur eine externe IP nötig



Probleme mit IPv4 nicht nur bei der Adressierung …

- Derzeitige IP-Version (**IPv4**) ist außerordentlich erfolgreich
- **Grund** liegt im geglückten Basisdesign:
  - Einsatz auch mit ganz neuen Hardware-Technologien möglich, wurde entwickelt vor Verbreitung der LAN-Technologie!
  - Funktioniert auch in Netzen, die um Größenordnungen schneller sind als die, für welche es ursprünglich konzipiert war
  - Hat enorme Zuwächse im globalen, heterogen Internet verkraftet

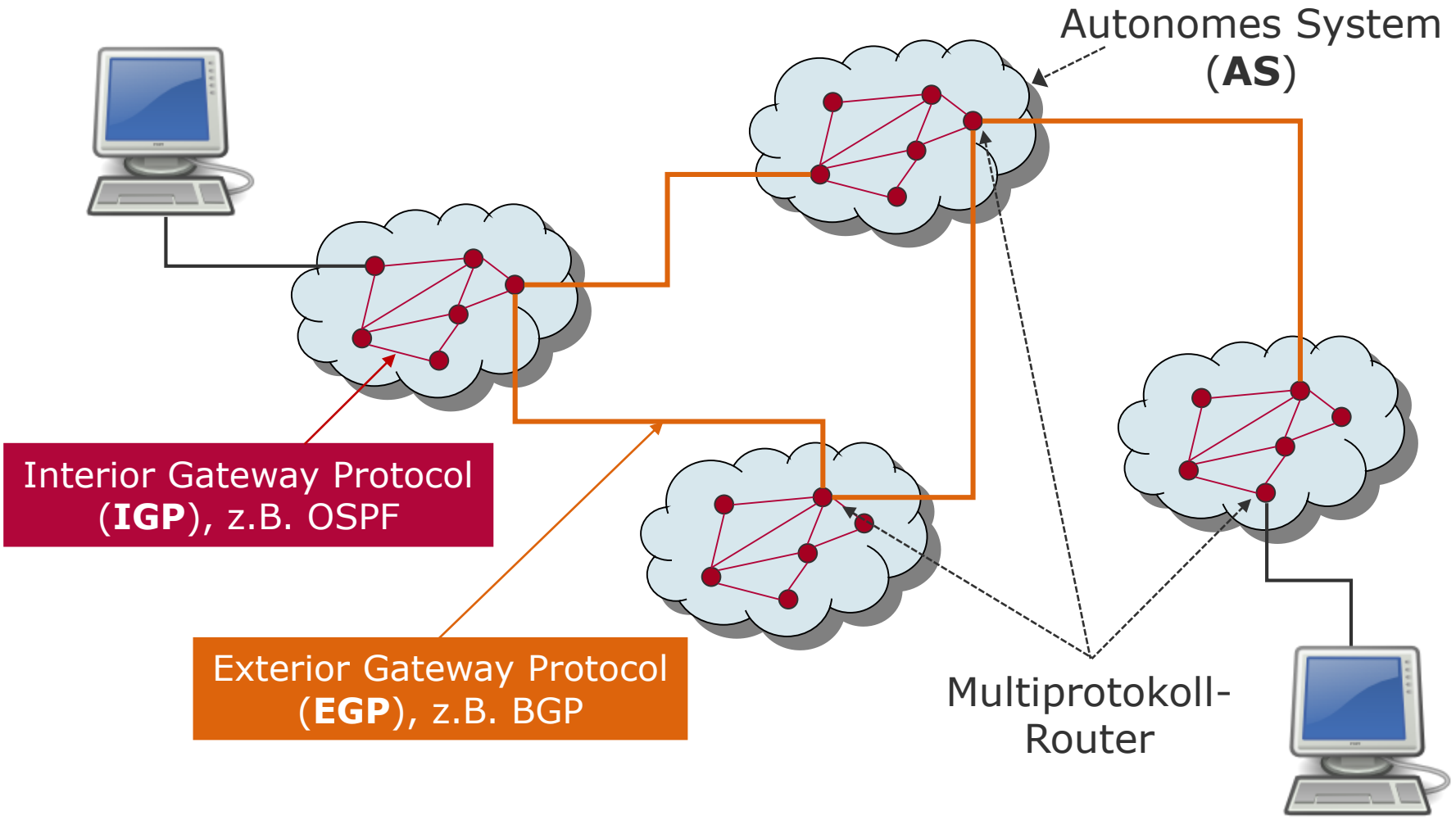
ABER ungebremstes Wachstum. Mobiles Internet und Entwicklung des Internet der Dinge (IoT) bringen IPv4 an seine Grenzen, z.B. fehlende Adressierungsmöglichkeiten
→ Umstieg auf Nachfolger IPv6 dringend nötig

openHPI-Kurs: Wie funktioniert das Internet?

Routing im Internet

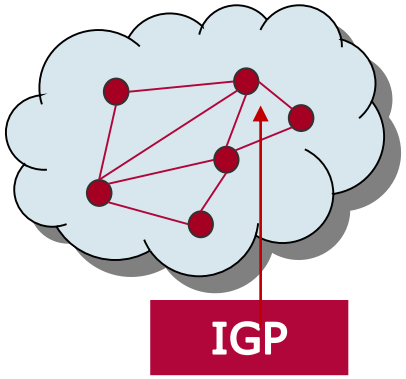
Prof. Dr. Christoph Meinel
Hasso-Plattner-Institut
Universität Potsdam

Routing im Internet



OSPF – Open Shortest Path First

Für das Routinginnerhalb Autonomer Systeme ist das **Interior Gateway Protokoll** – **IGP** – zuständig



IGP basiert auf dem **OSPF – Open Shortest Path First**:

- Standardisiert in RFC 2328, ursprünglich schon 1989 eingeführt als RFC 1131
- **Link-State-Routing**, basiert auf **Dijkstra-Algorithmus**
- Erlaubt
  - schnelle, dynamische Anpassung an Topologieveränderungen
  - hierarchisches Routing
  - Routing im LAN (Broadcast-Netzwerk) und WAN
- unterstützt unterschiedliche Metriken im Internet (kürzeste Routen, billigste Routen, …) und
- herstellerunabhängig

OSPF – Dijkstra-Algorithmus (1/2)

Netz wird als Graph G mit gewichteten Kanten modelliert

Ziel: Finde kürzeste Wege von Startknoten A zu allen Knoten von G

Vorgehensweise (1/2):

- Weise allen Knoten die beiden Eigenschaften **Distanz** und **Vorgänger** zu,
- führe zwei zunächst leere Knoten-Listen „**besucht**“ und „**gefunden**“ und
- initialisiere Distanz im Startknoten A mit 0 und in allen anderen Knoten mit ∞

OSPF – Dijkstra-Algorithmus (2/2)

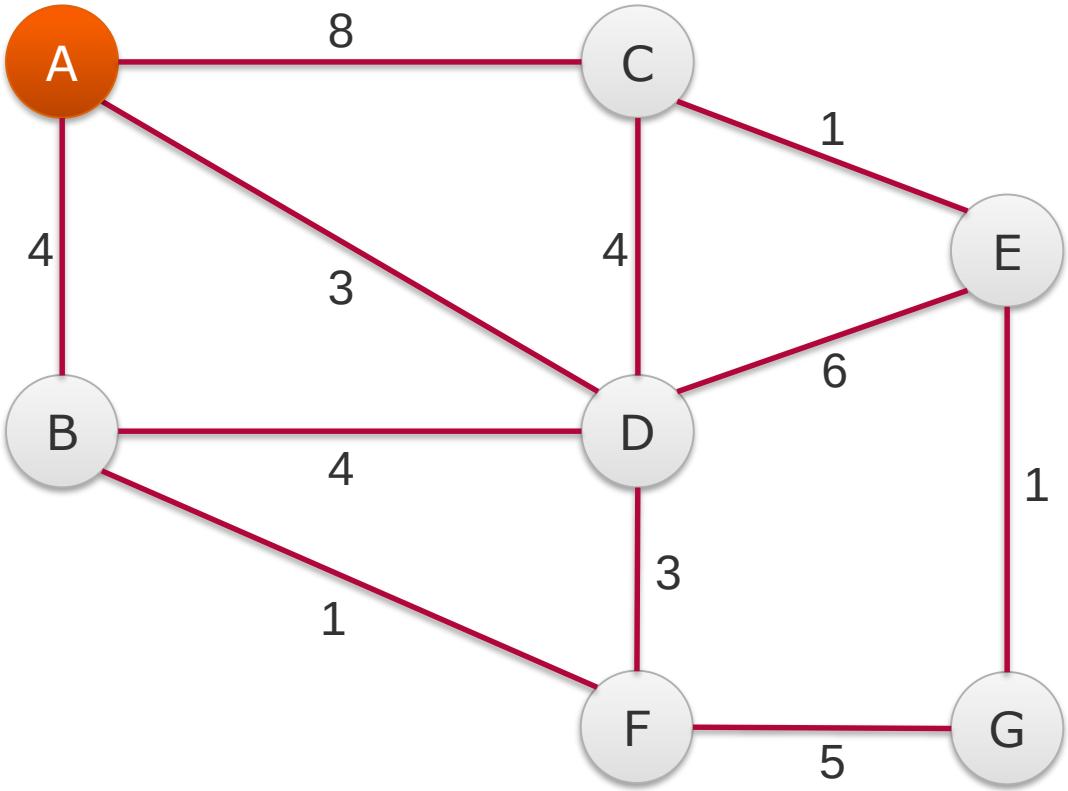
Ziel: Finde kürzeste Wege von Startknoten A zu allen Knoten von G.

Vorgehensweise (2/2):

Solange es Knoten gibt, die noch nicht gefunden sind,

- wähle einen Knoten mit **minimaler Distanz** aus und
  - speichere ihn Liste „**gefunden**" und
  - berechne für alle seine noch unbesuchten Nachbarknoten die Summe des jeweiligen Kantengewichtes und der aktuellen Distanz
    - ist Knoten noch nicht in besucht-Liste, wird er eingefügt
    - Ansonsten prüfe, ob gerade berechneter Wert kleiner ist als die in der besucht-Liste gespeicherte Distanz
      → aktualisiere Distanz und setze den aktuellen Knoten als Vorgänger (Update)

Beispiel: Dijkstra-Algorithmus (1/8)

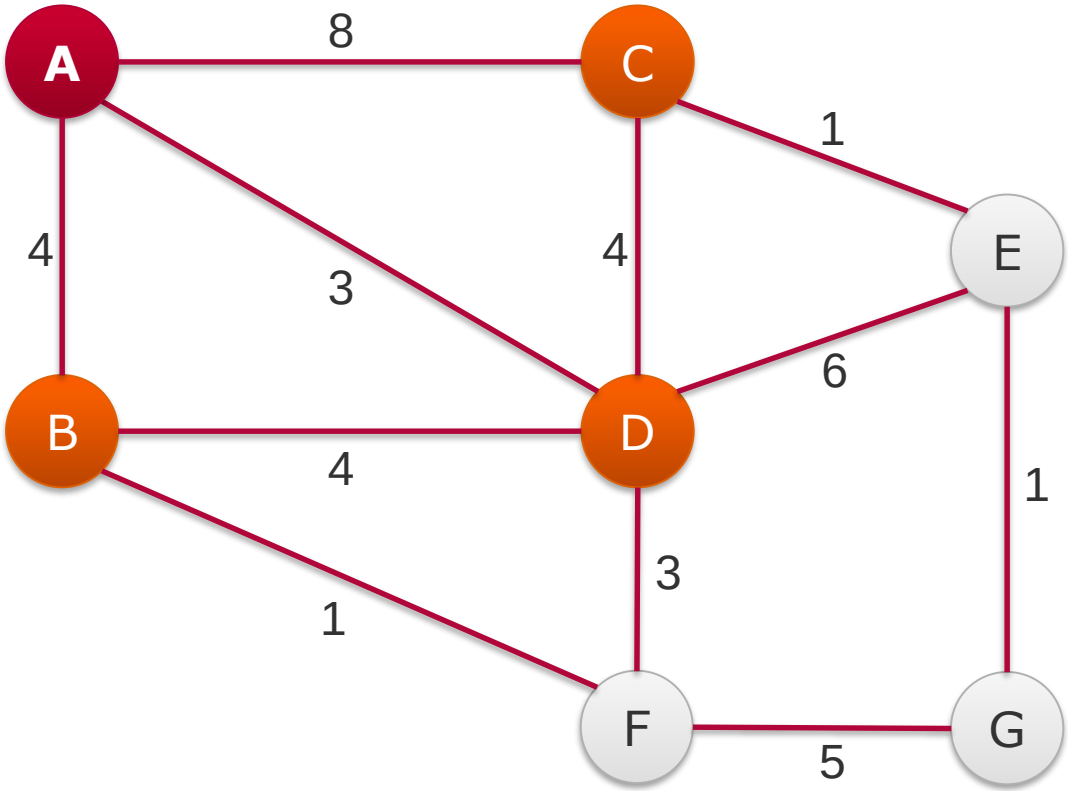


Ziel: Kürzeste Routen von A zu allen anderen Knoten finden

Wähle **Knoten A** aus besucht-Liste

Besucht	**A:0**
Kürzester Weg gefunden	

Beispiel: Dijkstra-Algorithmus (2/8)

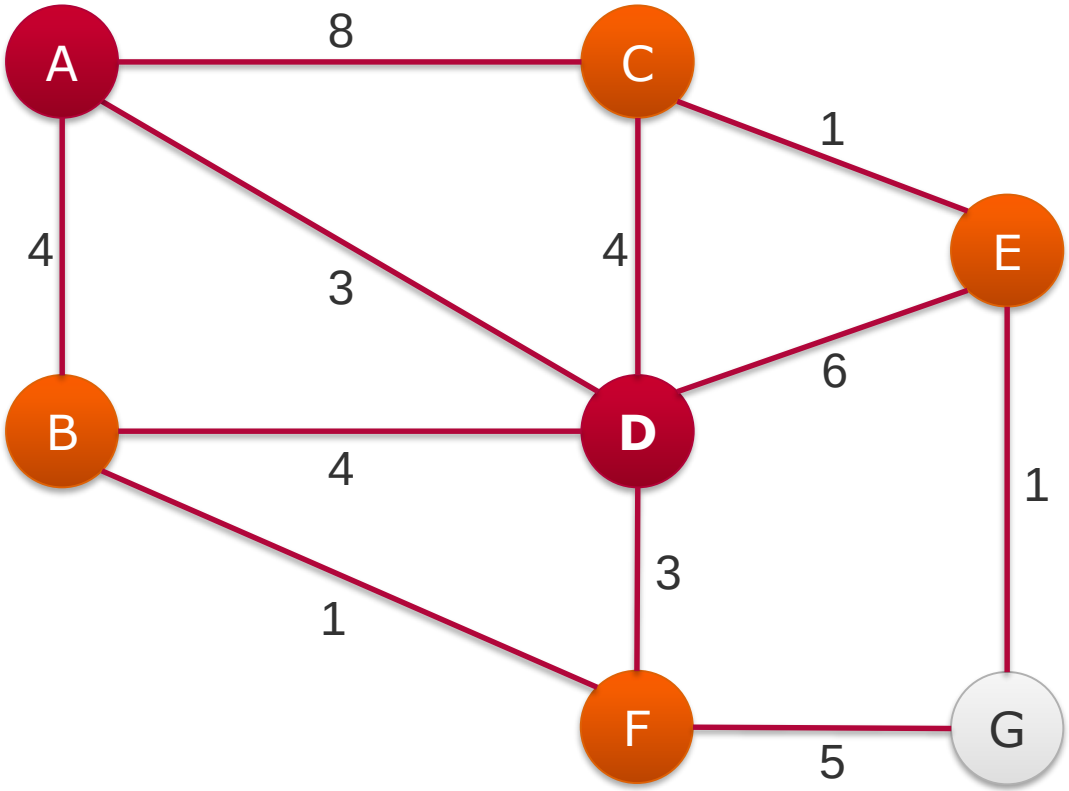


Ziel: Kürzeste Routen von A zu allen anderen Knoten finden

Wähle **Knoten D** aus besucht-Liste

Besucht	B:4, C:8, **D:3**
Kürzester Weg gefunden	A:0

Beispiel: Dijkstra-Algorithmus (3/8)



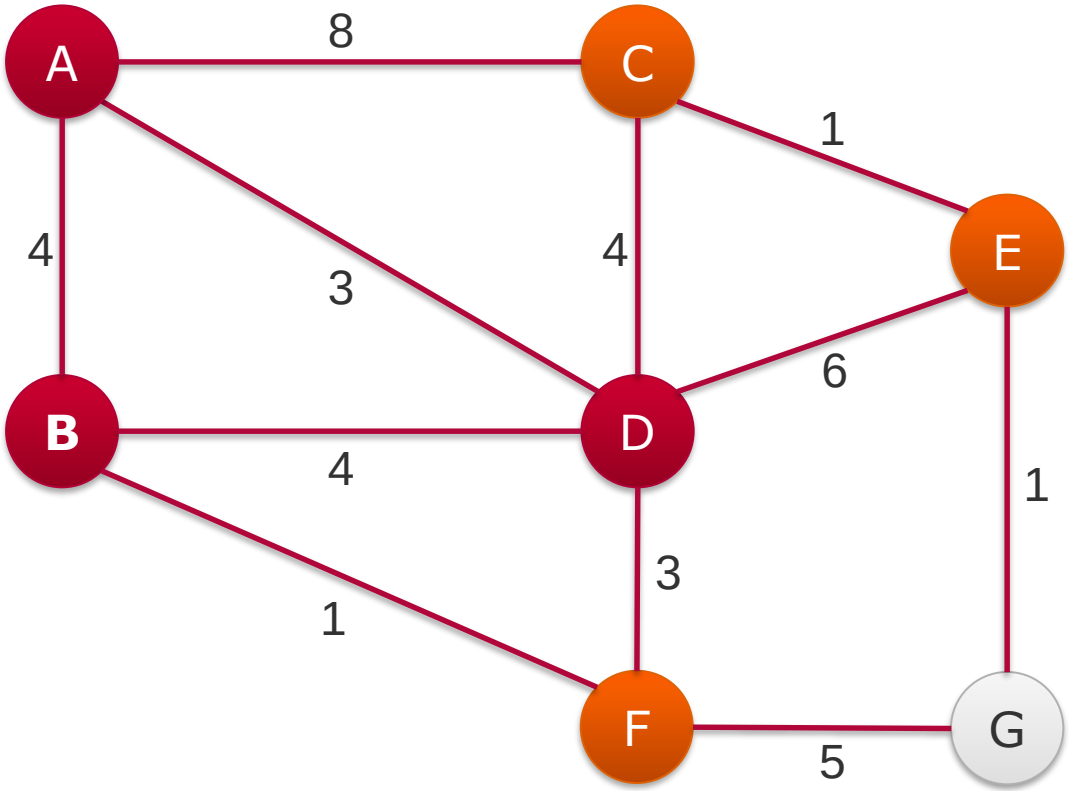
Ziel: Kürzeste Routen von A zu allen anderen Knoten finden

Update Distanz C (8 auf 7)

Wähle **Knoten B** aus besucht-Liste

Besucht	**B:4**, C:7, E:9, F:6
Kürzester Weg gefunden	A:0, D:3

Beispiel: Dijkstra-Algorithmus (4/8)



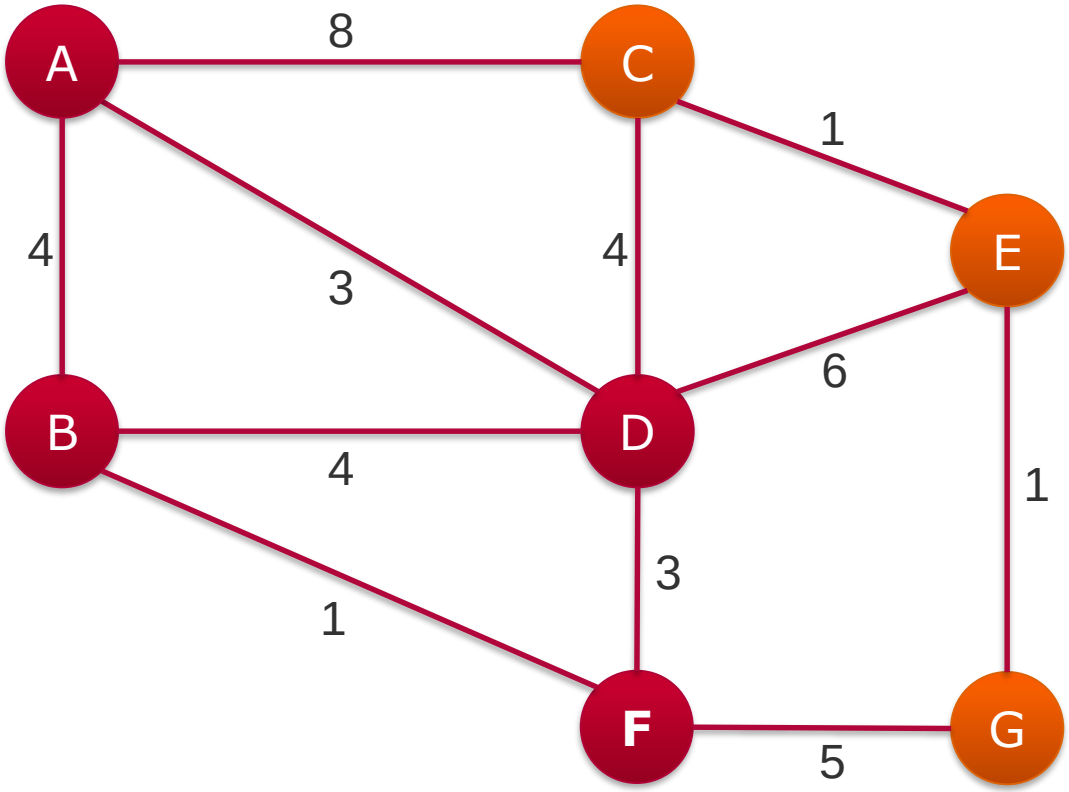
Ziel: Kürzeste Routen von A zu allen anderen Knoten finden

Update Distanz F (6 auf 5)

Wähle **Knoten F** aus besucht-Liste

Besucht	C:7, E:9, **F:5**
Kürzester Weg gefunden	A:0, D:3, B:4

Beispiel: Dijkstra-Algorithmus (5/8)

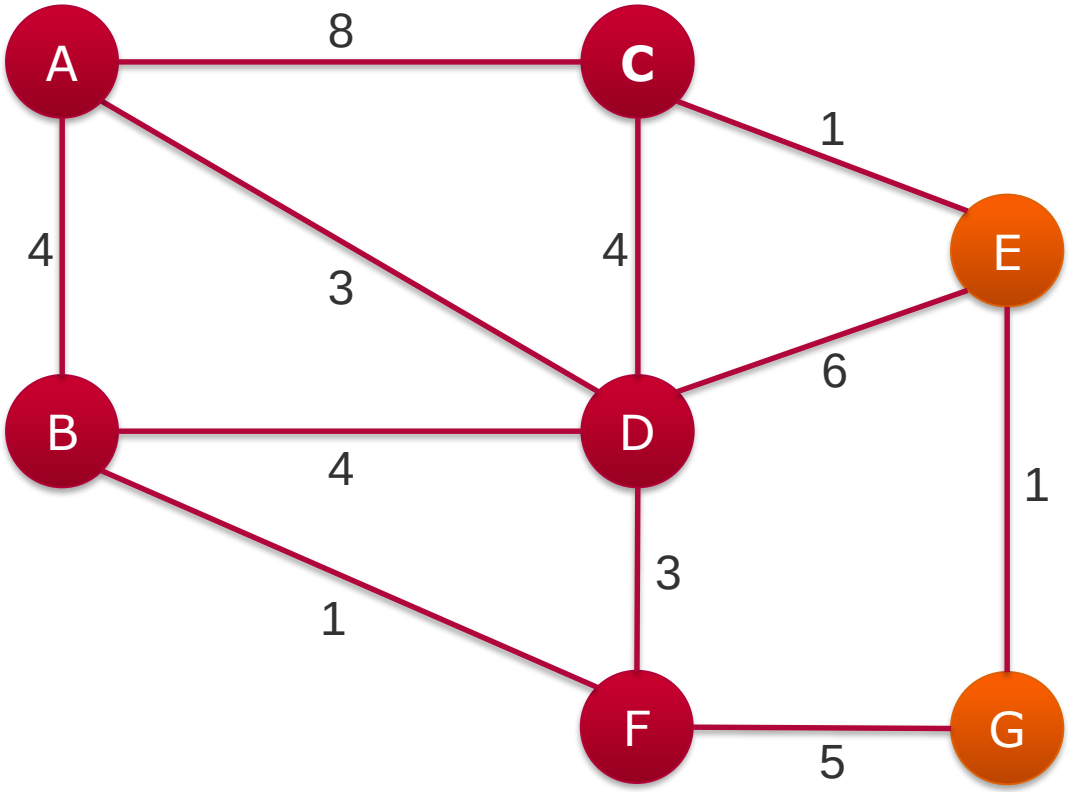


Ziel: Kürzeste Routen von A zu allen anderen Knoten finden

Wähle **Knoten C** aus besucht-Liste

Besucht	**C:7**, E:9, G:10
Kürzester Weg gefunden	A:0, D:3, B:4, F:5

Beispiel: Dijkstra-Algorithmus (6/8)



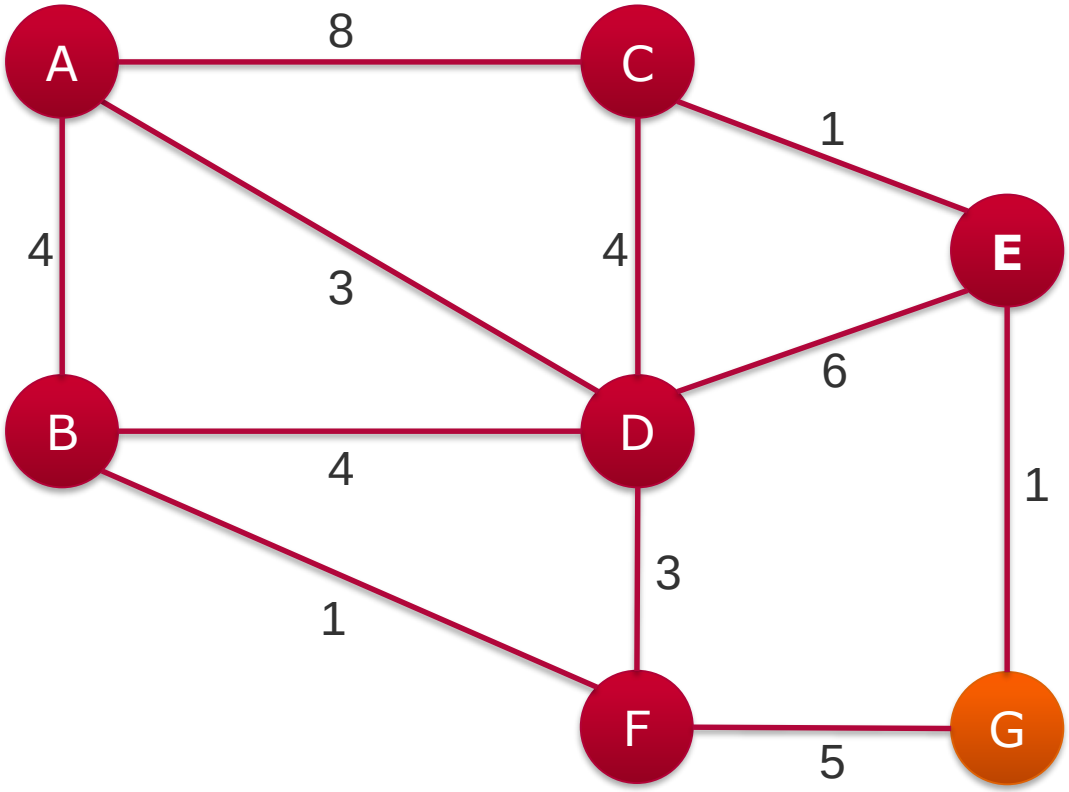
Ziel: Kürzeste Routen von A zu allen anderen Knoten finden

Update Distanz E (9 auf 8)

Wähle **Knoten E** aus Besucht-Liste

Besucht	E:8, G:10
Kürzester Weg gefunden	A:0, D:3, B:4, F:5, C:7

Beispiel: Dijkstra-Algorithmus (7/8)



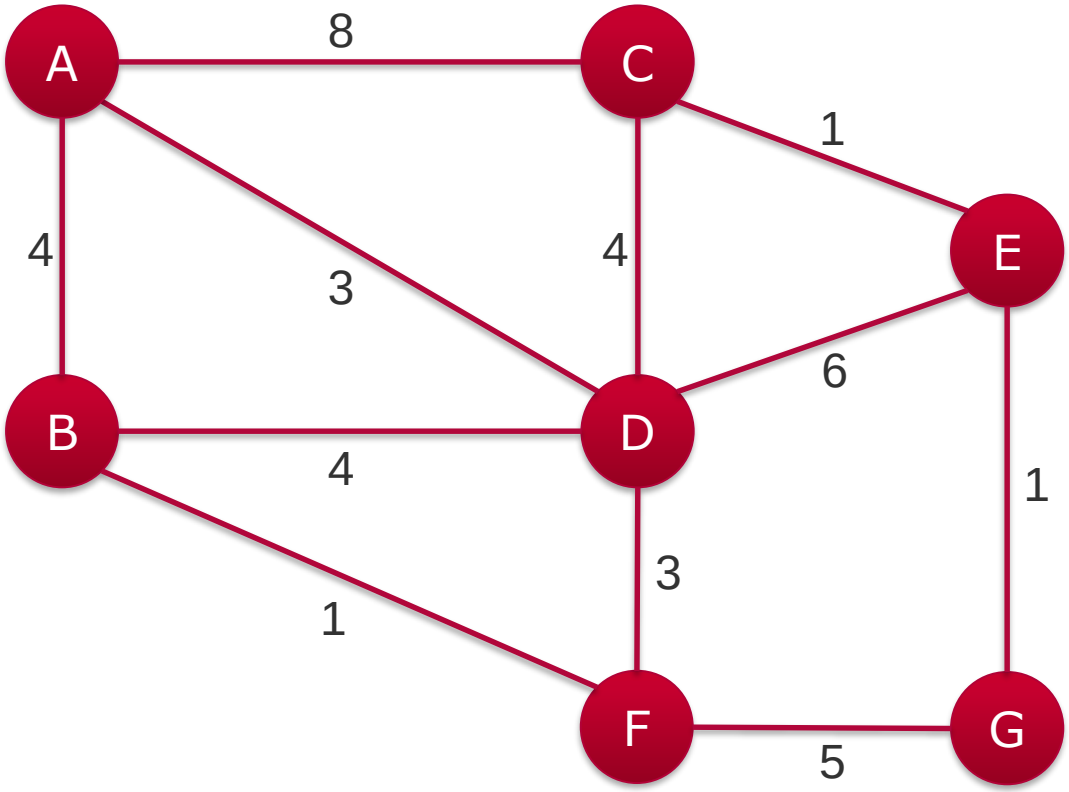
Ziel: Kürzeste Routen von A zu allen anderen Knoten finden

Update Distanz G (10 auf 9)

Wähle **Knoten G** aus besucht-Liste

Besucht	G:9
Kürzester Weg gefunden	A:0, D:3, B:4, F:5, C:7, E:8

Beispiel: Dijkstra-Algorithmus (8/8)



Ziel: Kürzeste Routen von A zu allen anderen Knoten finden

besucht-Liste leer
➔ **Algorithmus terminiert**

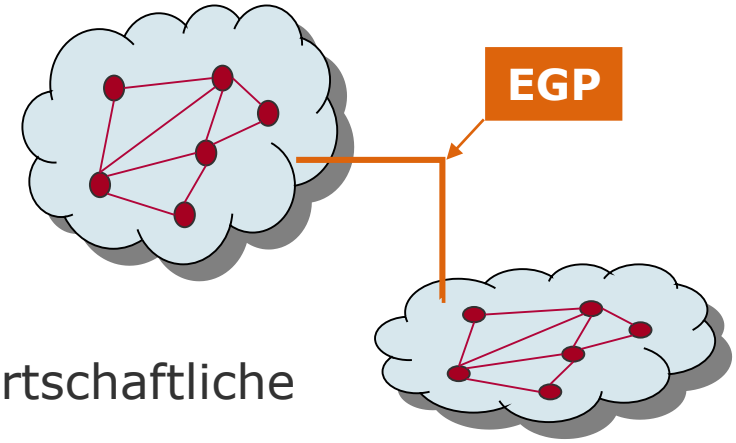
Besucht	
Kürzester Weg gefunden	A:0, D:3, B:4, F:5, C:7, E:8, G:9

OSPF – Open Shortest Path First

Nachrichtentypen für Kommunikation zwischen Routern

Hello	Wer sind meine Nachbarn?
Link State Update	Link-State-Aktualisierung an Nachbarn (Status, Metrik/Kosten)
Link State Ack	Bestätigung der Link-State-Aktualisierung
Database Description	Link-State-Pakete mit aktuellen Informationen des Senders
Link State Request	Link-State-Anforderung an Nachbarn zum Ermitteln der aktuellsten Verbindungsdaten

BGP – Border Gateway Protocol (1/2)



- Nur für AS-Grenz-Router(!), also für die Verbindung zwischen zwei Internet Service Providern
- Muss bestimmte Regelungen beachten
  - politische, sicherheitstechnische, wirtschaftliche
  - manuelle Konfiguration (Scripts)

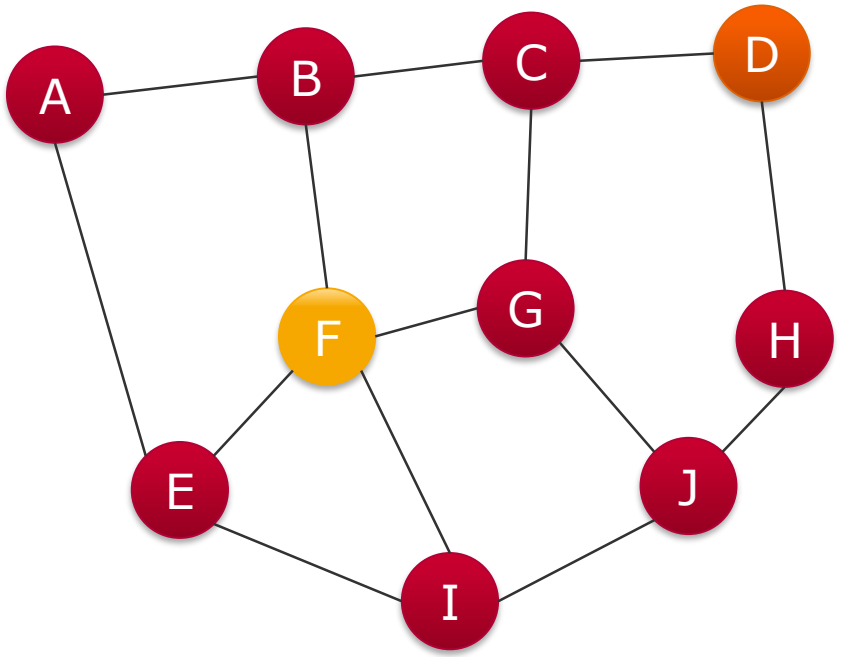
  (bei IGP → geht es „nur" um möglichst effiziente Paketzustellung)
- BGP-Router kommunizieren via TCP-Verbindung
  - zuverlässig, verbirgt Details der durchquerten Netze
- Grundsätzlich Distanzvektor-Routing
  - verwaltet Distanz UND zugehörigen Pfad
  - teilt Nachbarn den verwendeten Pfad mit

BGP – Border Gateway Protocol (2/2)

Beispiel:

Informationen, die **F** von seinen Nachbarn über den Router **D** erhält:

B:	B – C – D
G:	G – C – D
I:	I – J – H – D
E:	E – F – G – C – D



- F ermittelt z.B. kürzesten Pfad nach D,
  endgültige Routingentscheidung trifft Administrator durch Router-Regeln

openHPI-Kurs: Wie funktioniert das Internet?

IPv4-Datagramme

Prof. Dr. Christoph Meinel
Hasso-Plattner-Institut
Universität Potsdam

IPv4-Datagram

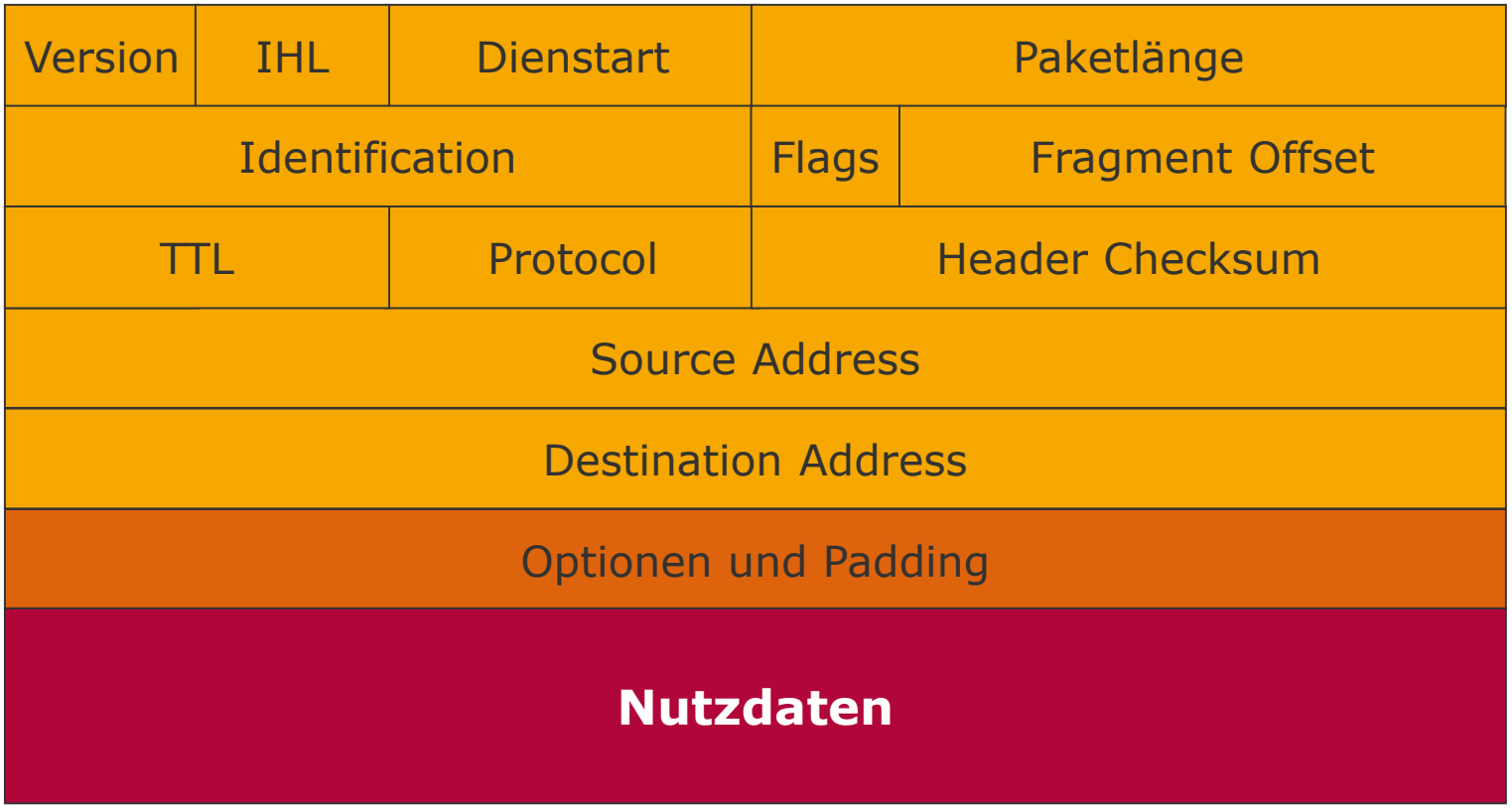
Datenpaket bei IP heißt **IP-Datagram**

- Maximale Länge von IPv4-Datagrammen: 65.535 Bytes
- IP-Datagramme bestehen aus einem Header und den eigentlichen Nutzdaten

- Im Header sind die Informationen versammelt, die zum Transport des Datagrams durch das Internet notwendig sind
- Die Nutzdaten sind die Daten, die durch das Internet transportiert werden sollen

IPv4-Header

Header (20 – 60 Byte)



IPv4-Headerfelder (1/2)

Version: ($0100_b = 4$)

Länge: Länge des IP-Headers
(min=5, max=15 → 32 Bit-Worte)

Dienstart: Priorität (0-7), D(elay), T(hroughput), R(eliability)

Paketlänge: 16 Bit

Identifikationsnummer (auch IP-ID genannt) einzelner Fragmente, evtl. vom Router vergeben

Flags: erstes Bit reserviert, DF/MF (*don't/more fragment)*

Fragment Offset: Offset des Fragments im originalen Datenpaket

Time to Live (TTL): Lebensdauer eines Pakets – wird bei jedem Hop vom Router dekrementiert

Protokoll: Nummer des übergeordneten Transportprotokolls

...

IPv4-Headerfelder (2/2)

...

Header Checksum: Prüfsumme;
muss bei jedem Hop neu berechnet werden

Source Address: Ursprung des Pakets (Sender)

Destination Address: Zieladresse des Pakets (Empfänger)

Options und Padding: Routing-Varianten, etc., z.B.

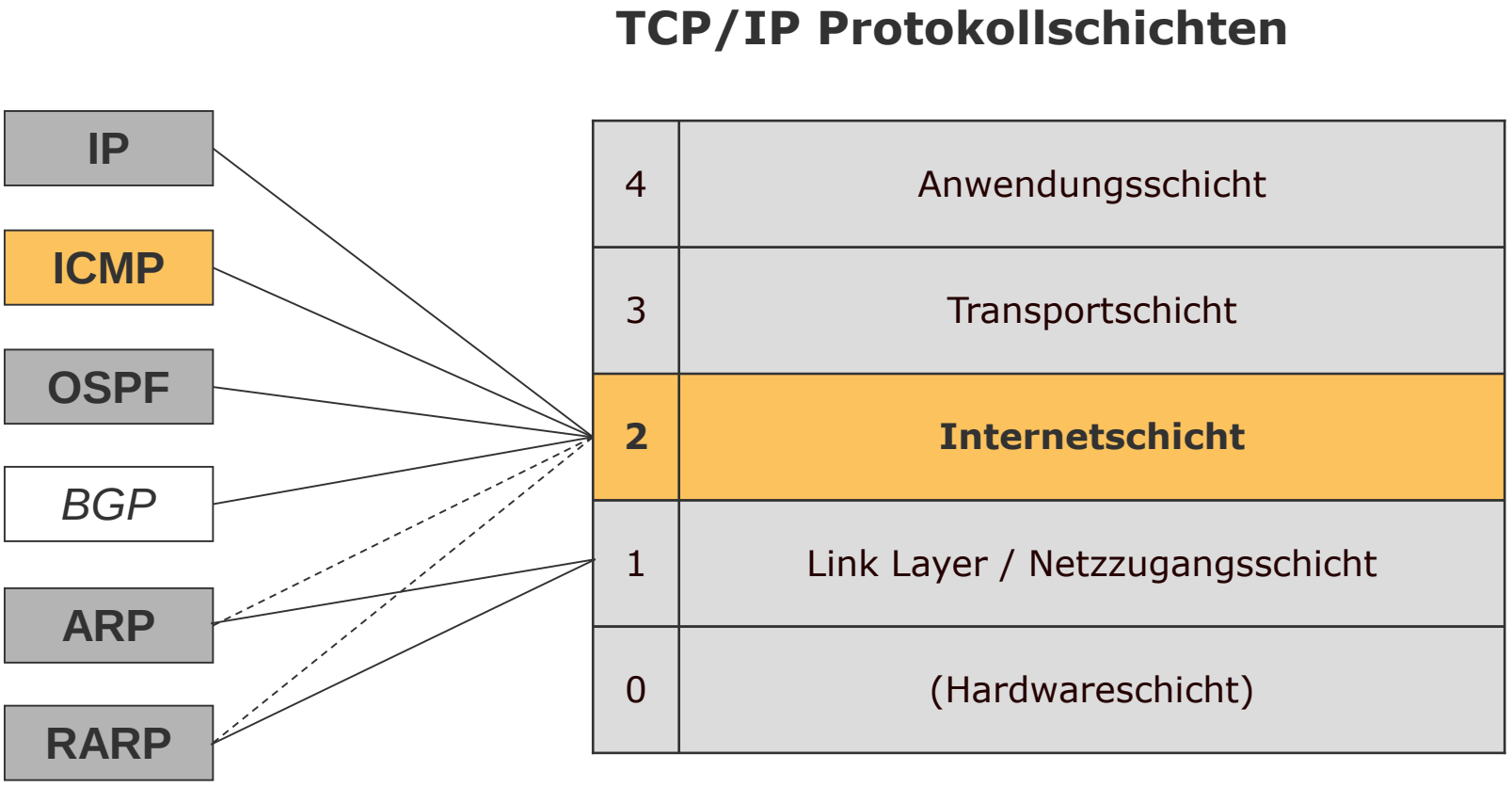
- **strict source routing**: Angabe des exakten Routing-Pfades
- **loose source routing**: Adressen der unbedingt zu besuchenden Router
- **record route**: jeder besuchte Router hängt seine Adresse an
- **timestamp**: jeder besuchte Router hängt seine Adresse und aktuelle Zeitinformation mit an

openHPI-Kurs: Wie funktioniert das Internet?

Weitere wichtige Protokolle – ICMP

Prof. Dr. Christoph Meinel
Hasso-Plattner-Institut
Universität Potsdam

Einordnung - ICMP



ICMP (1/5)

ICMP - Internet Control Message Protocol (RCF 792 / 1256)

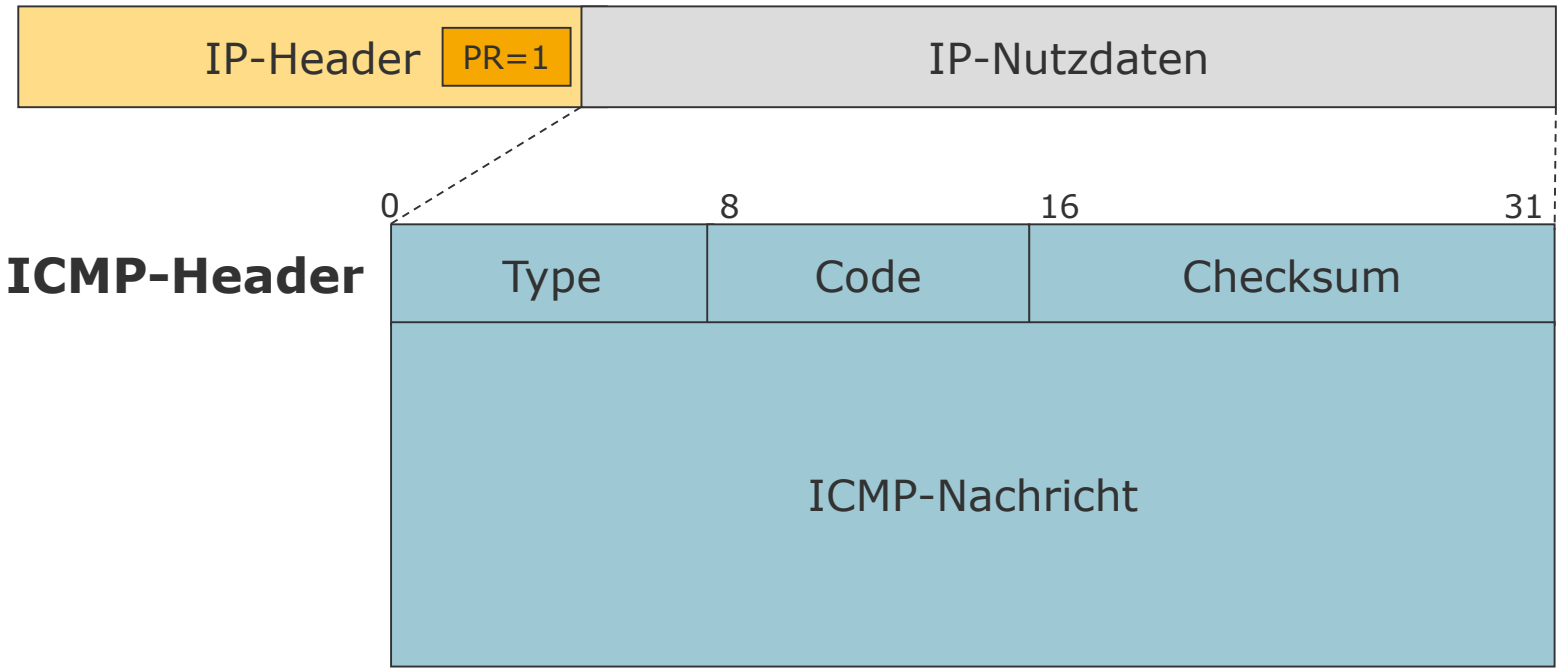
Aufgaben:

- Fehlerdiagnose, z.B. Erreichbarkeitstest via ping
- Aufzeichnung von Zeitmarken
- Erkennen abgelaufener Zeitmarken
- Verwaltung der Routing-Tabellen
- Flusskontrolle, um Überlast an Routern zu vermeiden
- Ermittlung der zulässigen MTU (Maximum Transfer Unit)

ICMP-Nachrichten werden gekapselt in IP-Datagramm
(Protokoll-Feld im IP-Header PR=1)

ICMP (2/5)

Datenformat von ICMP Nachrichten:



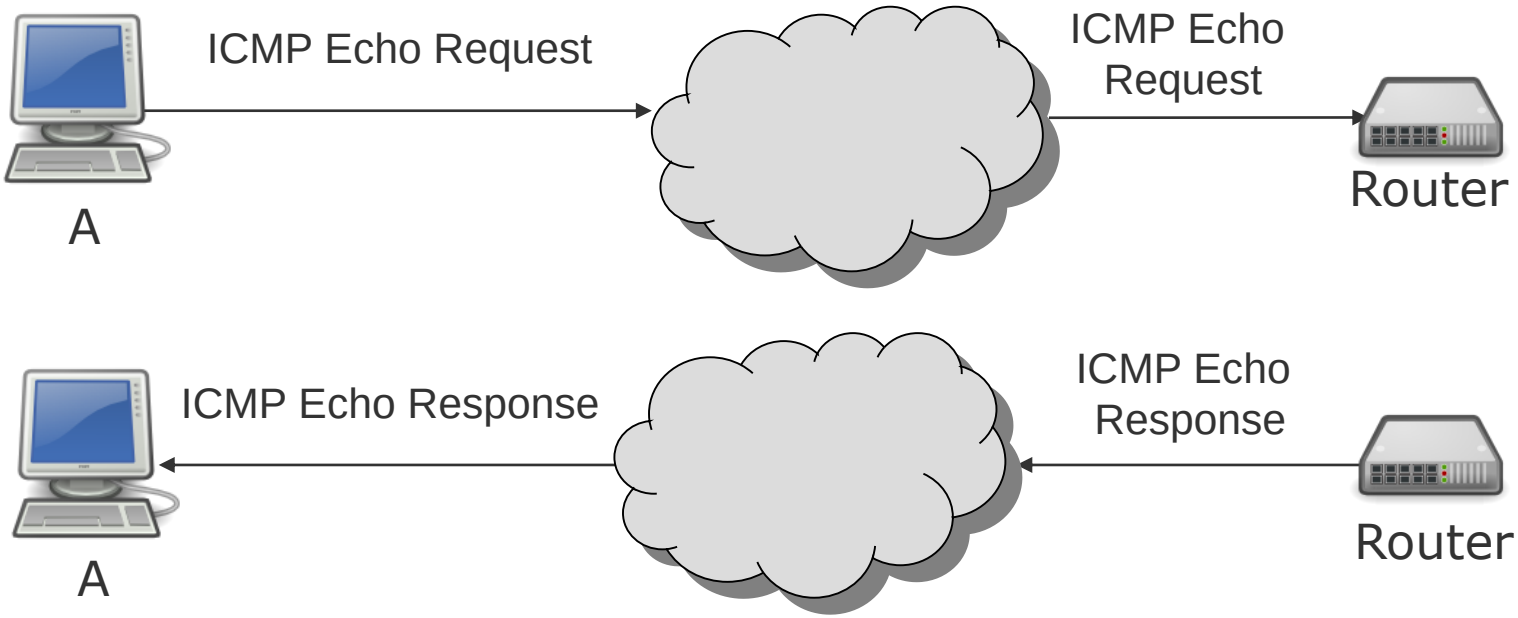
ICMP (3/5)

Nachrichtentypen

0	Echo Response
3	Ziel nicht erreicht
4	Senderate drosseln
5	Route ändern
8	Echo Request
9	Router Bekanntmachung
10	Suche nach einem Router
11	Lebenszeit eines IP-Datagramms überschritten
13	Timestamp-Anforderung
14	Timestamp-Antwort
17	Abfrage Subnetzmaske

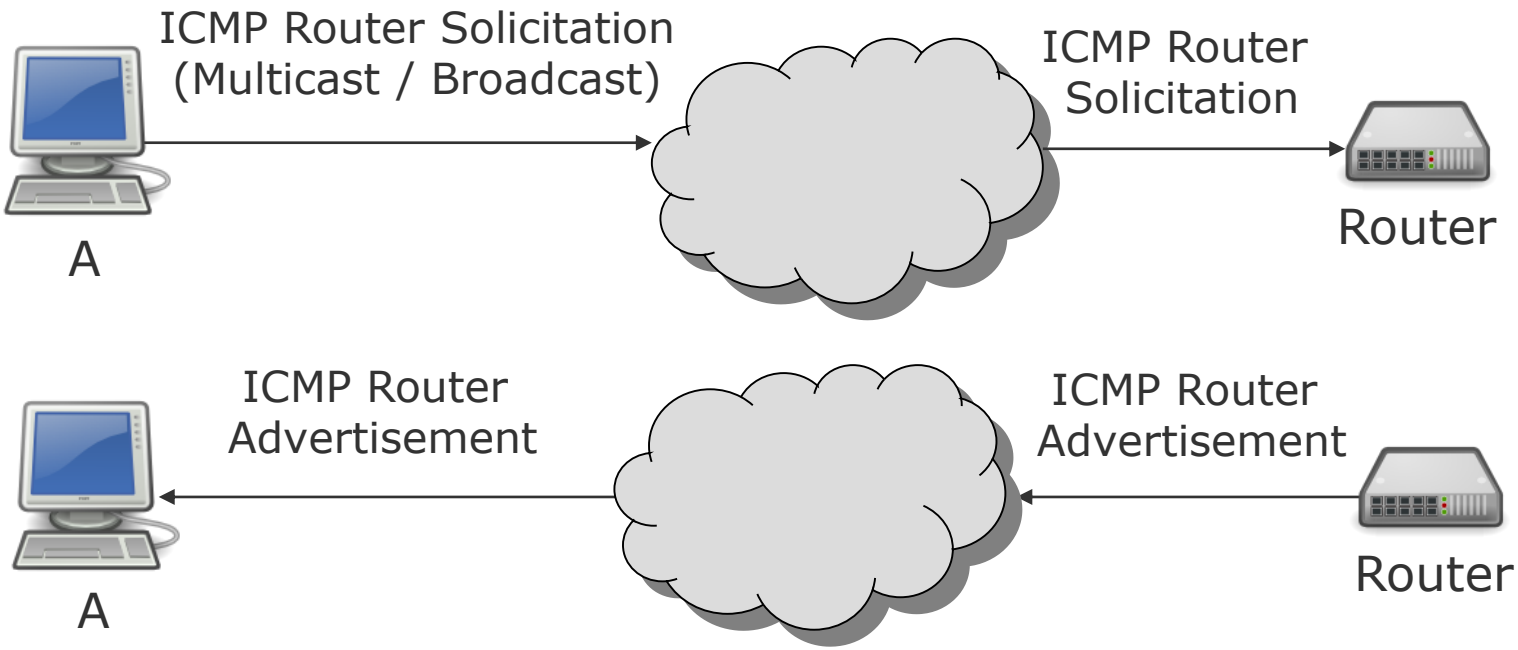
ICMP (4/5)

Erreichbarkeit testen



ICMP (5/5)

Router ermitteln (Router Solicitation)



Tool - ping

- Tool zum Testen der Erreichbarkeit von Hosts
- Nutzt die zuvor beschriebene Funktion ICMP Echo Request

```
sh-3.2$ ping 172.16.57.237
PING 172.16.57.237 (172.16.57.237): 56 data bytes
64 bytes from 172.16.57.237: icmp_seq=0 ttl=64 time=0.050 ms
64 bytes from 172.16.57.237: icmp_seq=1 ttl=64 time=0.084 ms
64 bytes from 172.16.57.237: icmp_seq=2 ttl=64 time=0.069 ms
64 bytes from 172.16.57.237: icmp_seq=3 ttl=64 time=0.074 ms
^C
--- 172.16.57.237 ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.050/0.069/0.084/0.012 ms
```

erreichbar

```
sh-3.2$ ping 192.168.2.1
PING 192.168.2.1 (192.168.2.1): 56 data bytes
Request timeout for icmp_seq 0
Request timeout for icmp_seq 1
Request timeout for icmp_seq 2
Request timeout for icmp_seq 3
^C
--- 192.168.2.1 ping statistics ---
5 packets transmitted, 0 packets received, 100.0% packet loss
```

nicht erreichbar

Tool - traceroute (1/2)

- Tool zum Ermitteln der Zwischenstationen zu einem Ziel-Host
- Nutzt die zuvor beschriebene Funktion ICMP Echo Request
  - Sendet mehrere ICMP-Pakete an den Ziel-Host
  - Setzt initiale TTL = 1 (Time to Live)
  - Zwischenstationen dekrementieren TTL bei jedem Hop und antworten bei **TTL = 0** mit „Time to live exceeded in transit”
    → Absender erfährt dadurch die Adresse der Zwischenstation, an der die TTL abgelaufen ist
  - Absender sendet erneut, jedoch mit um 1 erhöhter TTL
  - Wiederholen, bis Ziel erreicht
  - Somit werden alle Zwischenstationen inkrementell ermittelt

- Hinweis: da die Pakete nicht zwangsläufig die gleiche Route nehmen, sollte der Test mehrmals durchgeführt werden

Tool - traceroute (2/2)

Beispielaufruf für Route zu www.google.de

```
sh-3.2$ traceroute www.google.de
traceroute to www.google.de (172.217.17.227), 64 hops max, 52 byte packets
 1  172.16.56.1 (172.16.56.1)  2.984 ms  2.883 ms  2.837 ms
 2  10.0.4.1 (10.0.4.1)  1.917 ms  1.183 ms  1.234 ms
 3  141.89.226.129 (141.89.226.129)  2.028 ms  2.081 ms  2.111 ms
 4  10.6.2.1 (10.6.2.1)  2.019 ms  1.973 ms  2.027 ms
 5  xr-pot1-te1-3.x-win.dfn.de (188.1.33.149)  2.550 ms  2.798 ms  2.481 ms
 6  xr-pep1-te2-1.x-win.dfn.de (188.1.144.53)  2.811 ms  3.607 ms  2.707 ms
 7  cr-tub2-te0-0-0-7-4.x-win.dfn.de (188.1.146.30)  4.043 ms  3.505 ms  3.510 ms
 8  google.bcix.de (193.178.185.100)  132.496 ms  75.668 ms  77.489 ms
 9  209.85.249.182 (209.85.249.182)  11.881 ms  11.394 ms  10.708 ms
10  72.14.232.25 (72.14.232.25)  10.808 ms  10.727 ms  10.672 ms
11  ber01s08-in-f3.1e100.net (172.217.17.227)  10.023 ms  10.181 ms  10.162 ms
```

Zwischen-Stationen

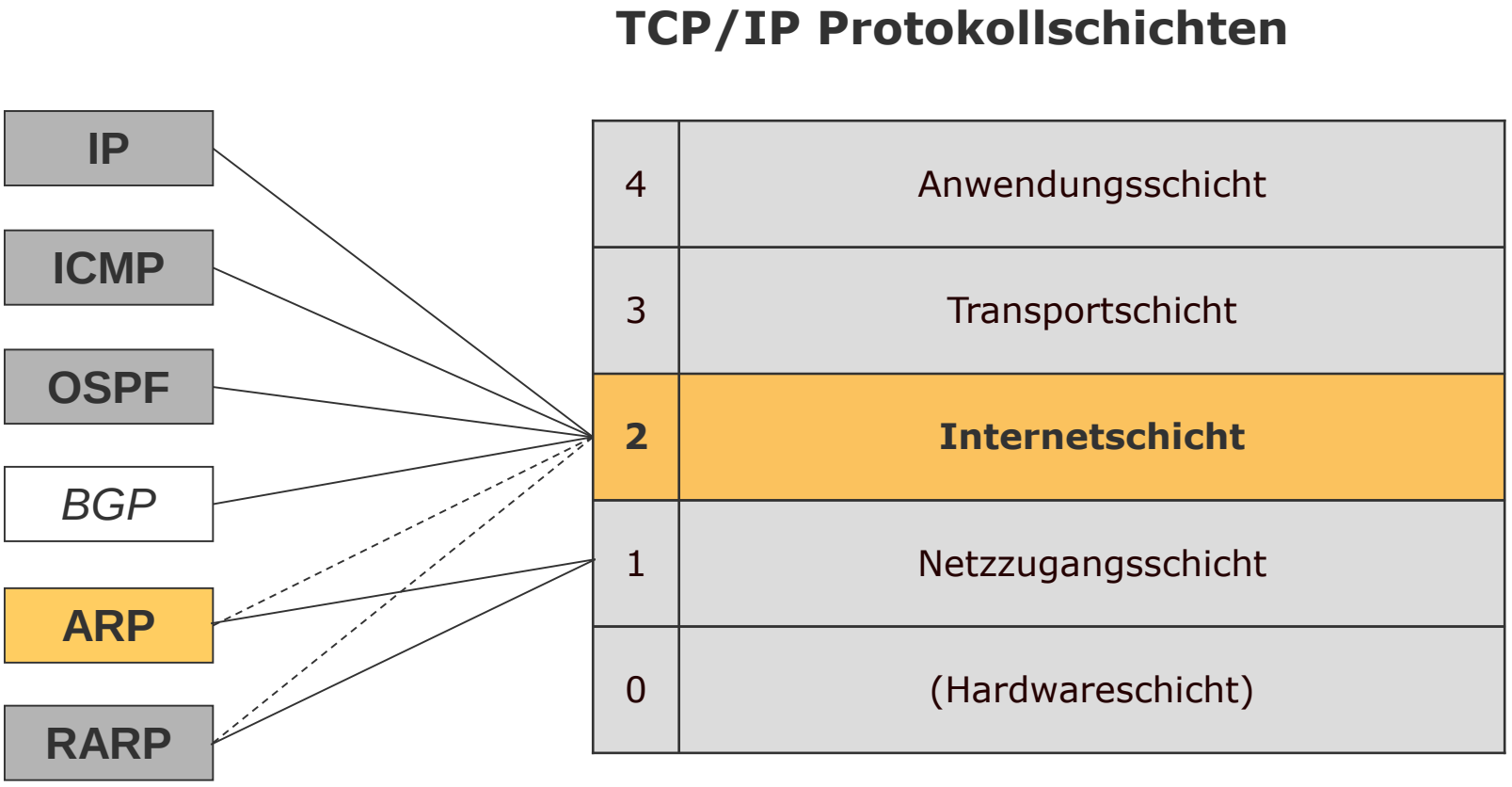
Ziel

openHPI-Kurs: Wie funktioniert das Internet?

Weitere wichtige Protokolle – ARP

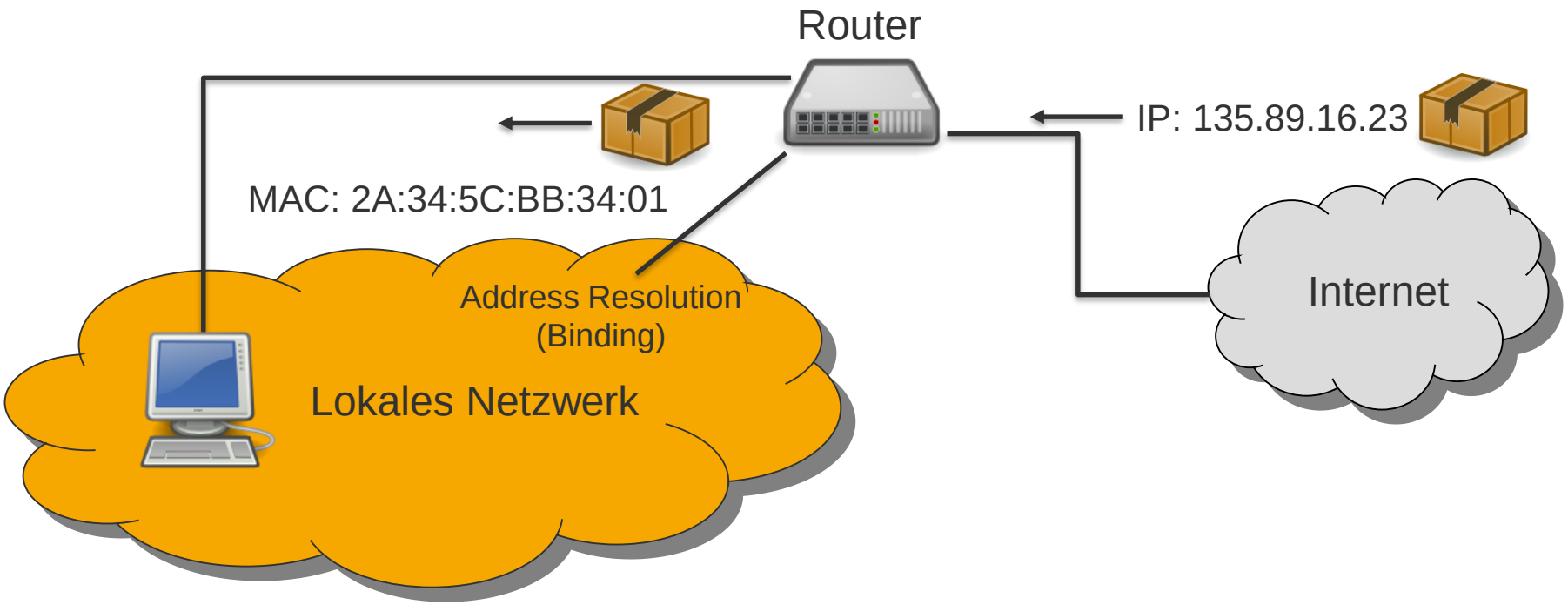
Prof. Dr. Christoph Meinel
Hasso-Plattner-Institut
Universität Potsdam

Einordnung - ARP



ARP – Address Resolution Protocol (1/3)

Router muss im lokalen Netzwerk IP-Adressen in Netzwerkadressen (z.B. Ethernet-Adresse) umwandeln



ARP – Address Resolution Protocol (2/3)

Mögliche Methoden:

1. **Tabellensuche**
   - Router verfügt über interne Adresstabelle zur Übersetzung
   - nur für kleine Netzwerke praktikabel (Durchsatz)
2. **Direkte Berechnung**
   - Router kann Adresse berechnen: f (IP-Adresse) = HW-Adresse
   - Voraussetzung: HW-Adresse lässt sich so festlegen, dass Berechnung möglich ist
3. **Nachrichtenaustausch**
   - Router sendet Anfrage zur Adressauflösung einer IP-Adresse
   - zentral: ARP-Server antwortet mit HW-Adresse
   - dezentral: Router Broadcast → Rechner verwalten eigene IP-Adresse und antworten, wenn IP-Adresse erkannt ist

ARP – Address Resolution Protocol (3/3)

ARP – Nachrichtenformat

- ARP-Anfrage: IP-Adresse + Übersetzungsaufforderung
- ARP-Antwort: IP-Adresse + HW-Adresse

Problem:

- Ineffizient, wenn vor jedem Datentransfer erst ARP-Anfrage initiiert werden muss!

Lösung:

- Verwendung eines Zwischenspeichers für bereits erfolgte IP-/HW Adressübersetzungen (**ARP-Cache**)

ARP – Einträge einsehen

Praxis: Einsicht der lokalen ARP-Konfiguration mit dem Tool „arp“

```
sh-3.2$ arp -a
? (172.16.56.1) at cc:4e:24:d0:f1:80 on en0 ifscope [ethernet]
? (172.16.56.57) at 0:c2:c6:77:39:68 on en0 ifscope [ethernet]
? (172.16.56.68) at 4c:34:88:16:8a:dc on en0 ifscope [ethernet]
? (172.16.56.81) at a0:e4:53:c5:8a:b5 on en0 ifscope [ethernet]
? (172.16.56.208) at 60:3:8:9b:11:b8 on en0 ifscope [ethernet]
? (172.16.57.60) at c8:bc:c8:c3:a3:2a on en0 ifscope [ethernet]
? (172.16.57.61) at b0:df:3a:10:b5:de on en0 ifscope [ethernet]
? (172.16.57.65) at 34:13:e8:30:87:84 on en0 ifscope [ethernet]
? (172.16.57.66) at 30:a8:db:8e:2c:88 on en0 ifscope [ethernet]
? (172.16.57.82) at 80:19:34:47:e3:5c on en0 ifscope [ethernet]
```

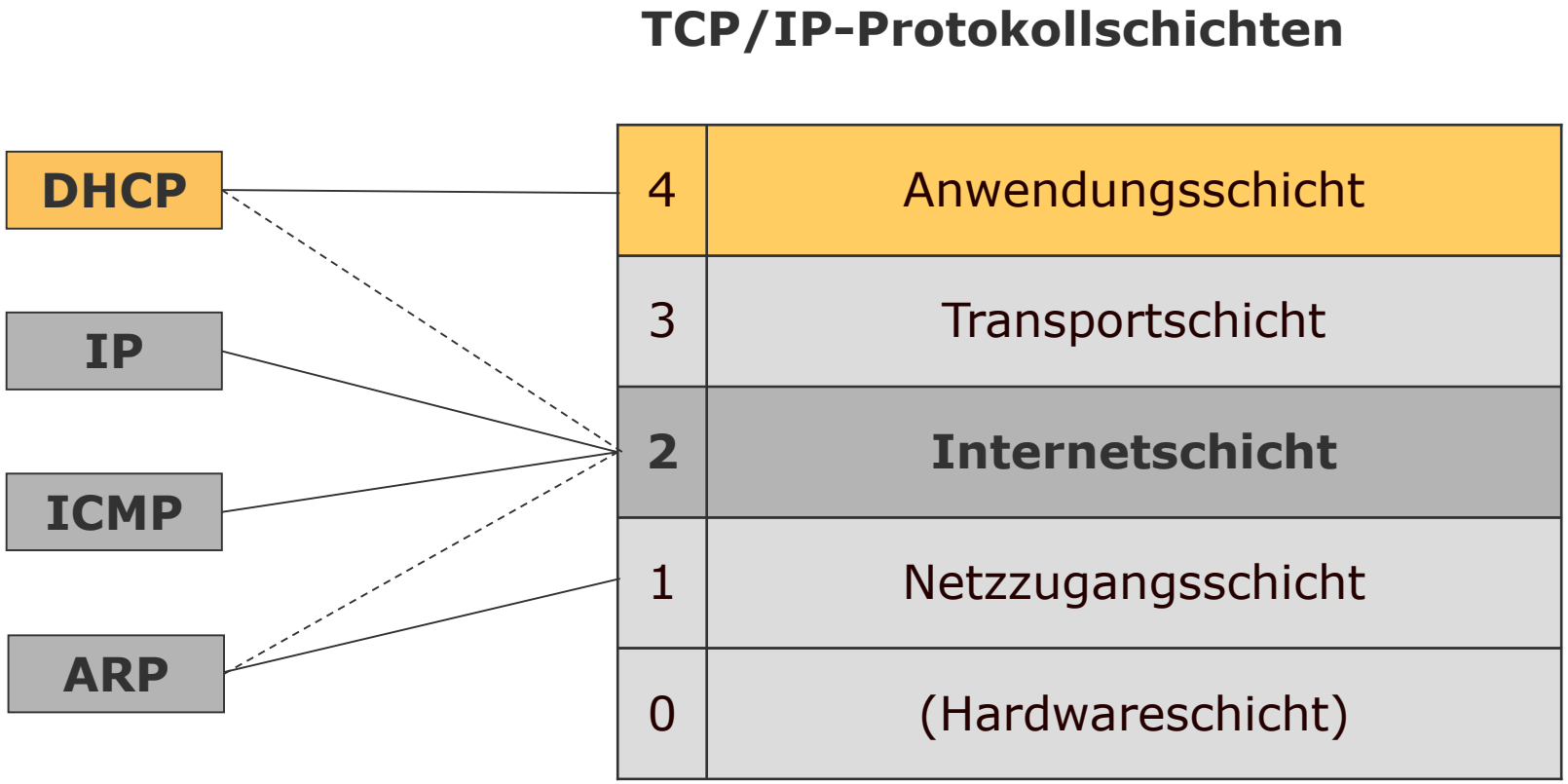
IP-Adresse MAC-Adresse

openHPI-Kurs: Wie funktioniert das Internet?

Weitere wichtige Protokolle - DHCP

Prof. Dr. Christoph Meinel
Hasso-Plattner-Institut
Universität Potsdam

Einordnung
DHCP – Dynamic Host Configuration Protocol



DHCP – Dynamic Host Configuration Protocol (1/3)

Problem:

- Manuelle Konfiguration von Netzwerkschnittstellen ist aufwendig

Lösung: (für IPv4)

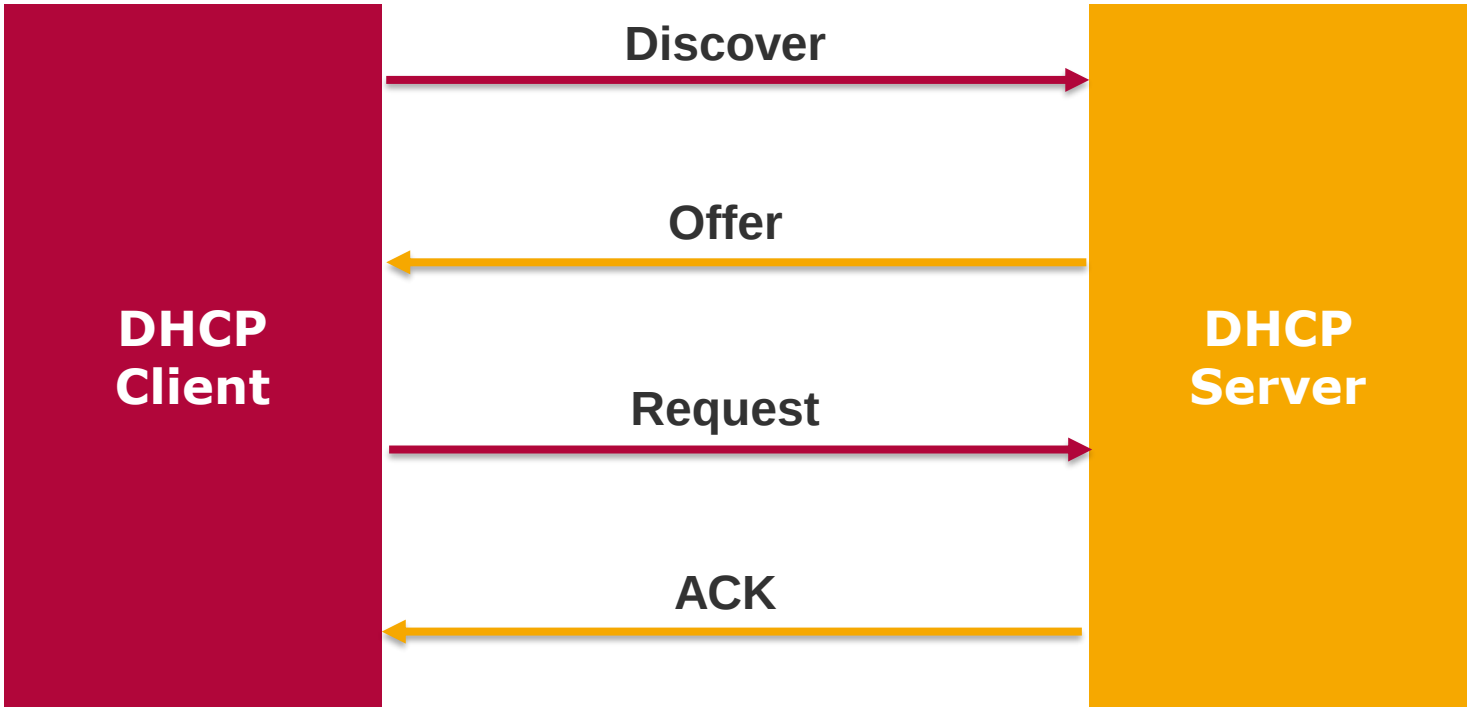
- **DHCP** (**Dynamic Host Configuration Protocol)**
  - Zuteilung einer (dynamischen) **IP-Adresse**
  - Zuteilung von **Hostnamen**
  - Information über **Default Gateway**
  - Information über **zuständigen DNS Server** (Domain Name Service)
- Operiert über UDP (Protokoll der Transportschicht)
- Arbeitet zustandsbehaftet (stateful)
  - der jeweilige DHCP-Server weiß, welcher Host welche Konfiguration benutzt und führt darüber Buch

DHCP – Dynamic Host Configuration Protocol (2/3)

Grundlegende Funktionsweise

1. **DHCP – Discover**
   - Client sendet Broadcast an DHCP-Server
   - Signalisiert Bedürfnis, eine IP-Adresse zu erhalten
   - Paket-Ziel: 255.255.255.255
   - Paket-Quelle: 0.0.0.0 (da noch keine IP zugewiesen)
2. **DHCP – Offer**
   - Server antwortet mit Menge von Adressen, welche als Vorschlag für den Client gelten
3. **DHCP – Request**
   - Client antwortet mit einer aus der Menge ausgewählten Adresse
4. **DHCP – ACK**
   - Server bestätigt Vergabe der Adresse
   - Kann ggf. Netzinformationen enthalten (z.B. Standard Gateway)

DHCP – Dynamic Host Configuration Protocol (3/3)



DHCP-Parameter überprüfen

Praxis

Einsicht der IP-Konfiguration je nach Betriebssystem mit dem Tool ipconfig bzw. ifconfig möglich

```
sh-3.2$ ifconfig
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        ether 00:11:22:33:44:55
        inet6 fe80::d2a6:37ff:feeb:3b63%en0 prefixlen 64 scopeid 0x4
        inet 172.16.57.237 netmask 0xfffffc00 broadcast 172.16.59.255
        nd6 options=1<PERFORMNUD>
        media: autoselect
        status: active
```

openHPI-Kurs: Wie funktioniert das Internet?

Internet als Angriffsziel

Prof. Dr. Christoph Meinel
Hasso-Plattner-Institut
Universität Potsdam

Erfolg des Internet macht es zum beliebten Angriffsziel

Internet und damit verbundenen Kommunikationstechnologien sind Schlüsseltechnologien der modernen Informationsgesellschaft

- Industrie und Wirtschaft machen reichlich Gebrauch von den Potenzialen der heutigen Kommunikationsmöglichkeiten; Popularität der vielen attraktiven kommerziellen und privaten Anwendungen macht das Internet zu einem globalen Markt

Kommerzielle Nutzung des Internets hat zu extremer Zunahme von Computer- und Cyber-Verbrechen geführt. Gefährdet sind:

- Computersysteme selbst
- Verbindungen und Systeme im Internet
- Abgeschlossene Intranets in Unternehmen

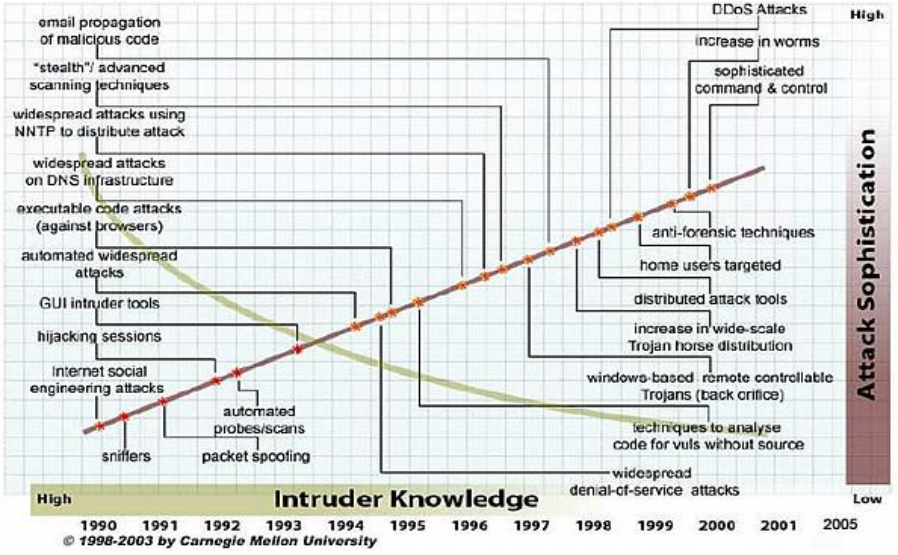
Internet ist ein einfaches Angriffsziel

Millionen miteinander verbundener Netzwerke bieten weltweit Milliarden von Zugangsstellen für Cyberkriminelle

Angriffe im Internet werden begünstigt durch

- Viele Design-Fehler in den Kommunikationsprotokollen
- Große Anzahl von Sicherheitslücken
- Einfacher Zugang zu Hacking-Tools
- Einfache Nutzung von Hacking-Tools
- …

Ergebnis: Die Raffinesse der Angriffe steigt, während die von Angreifern erforderlichen Kenntnisse sinken



Systematische Probleme im Bereich der Internetsicherheit

- Internet begann als Forschungsprojekt
  - kleine Forschungsgemeinschaft
  - überschaubares und vertrauenswürdiges Klientel
- Sicherheit war kein primärer Gesichtspunkt bei der Entwicklung von Internet-Protokollen
  - Typischer Satz in Internet-Standards/RFCs – Request for Comments:
    "Sicherheitsfragen sind in diesem Memo nicht behandelt"
- Internet ist Verbund unabhängiger Rechnernetze, keine Internet-Behörde und keine staatliche Stelle kann das komplette Internet kontrollieren
- …

Internet – viele Angriffspunkte

- Internet und internetbasierte Systeme bieten viele verschiedene Angriffspunkte und Risiken
  - bei den benutzten Computersystemen, z.B. PC, Laptop, Smartphone, Tablet, …
  - bei der Anbindung an das Internet, in den Netzwerken und Zwischensystemen des Internets und
  - innerhalb von privaten Netzwerken, Unternehmen und Institutionen, die die Internet-Technologie in ihren Intranets nutzen
- Angriffe über das Internet sind einfach zu realisieren, mit relativ geringem Entdeckungsrisiko und schwer zu verfolgen
- Rasche Entwicklungszyklen bei Hardware- und Softwaretechnik hinterlassen Tausende von Schwachstellen und Angriffspunkten

  → **99% der Angriffe nutzen bekannte Schwachstellen aus !**

Verbesserte Chancen für Intrusion, also Einbrüche über das Internet

- Komplexität des Internet, seiner Protokolle und Anwendungen sowie Abhängigkeit vom Internet in Wirtschaft und Gesellschaft steigt ständig
- Angreifer sind gut vorbereitet und organisiert. Ihre Werkzeuge (Hacking Tools) sind immer ausgereifter, einfach zu bedienen und unterstützen massive Angriffe
- Vielfältiger Mangel an
  - Bewusstsein für Informationssicherheit
  - qualifiziertem Informationssicherheitspersonal, System- und Netzwerk-Administratoren
  - Regulierungen im globalen Internet und Machtmittel zur Durchsetzung gesetzlicher Vorgaben

Angriffspunkte beim IPv4-Protokoll

Internetprotokoll IPv4 ist äußerst erfolgreich bei der Kommunikation im Internet, hat jedoch auch Schwächen. So bietet IPv4 selbst z.B.

- keine Mechanismen zur Verschlüsselung
- keine Mechanismen zur Authentifizierung, also zur Feststellung der Identität des Senders

Deshalb kann beim Transport von IPv4-Paketen im Internet

- jeder beteiligte Rechner auf der Transportroute das Paket lesen und, schlimmer noch,
- das Paket manipulieren, bevor er es weiterschickt

Es ist leicht, IPv4-Pakete unter falschem Absender zu verschicken, indem im IP-Header eine falsche Absenderadresse angegeben wird

→ sehr beliebte Angriffsmethode **IP-Spoofing**

Angriffspunkte beim IPv4-Protokoll
Zum Beispiel: IP-Spoofing

Vortäuschung eines falschen Absenders, also das Versenden von IPv4-Paketen mit falscher Absender-IP-Adresse heißt → **IP-Spoofing**

Anwendungen:

- Überlistung von Sicherheitsüberprüfung durch Firewalls
- Erlaubt (fast) anonyme Angriffe auf Fremdsysteme, z.B.
  - Erzeugung von Überlast „**Denial of Service**" (DoS)
  - unberechtigte Authentifizierung per IP-Adresse

Tools zum IP Spoofing:

- **Nmap, hping**

Die genannten Applikationen sind zur Überprüfung der Netzwerksicherheit gedacht.

Jeglicher Angriff auf Geräte ohne Berechtigungen ist verboten und fällt unter den Straftatbestand der Computersabotage (§ 303b StGB)!

Maßnahmen für mehr Sicherheit

- Einsatz von **IPsec – Internet Protocol Security**
  - Protokollsuite zur Absicherung von IP-Kommunikation
  - Bietet für jedes IP-Paket **Verschlüsselungs-** und **Authentifizierungsmechanismen**
- Auch möglich: Verlagerung der Sicherheitsprüfung auf **höhere Schichten** im Protokollstapel
  - Auf der Transportschicht z.B.
    - bei TCP: Überprüfung der Sequenznummer
  - Auf der Anwendungsschicht z.B.
    - Paketfilter: Router filtert eingehende IP-Pakete aus externen Netzen, die vorgeblich aus internem Netz stammen
- Einsatz von → **VPN – Virtual Private Network**