



---

# Rechnernetze II

SoSe 2020

Roland Wismüller  
Betriebssysteme / verteilte Systeme  
roland.wismueller@uni-siegen.de  
Tel.: 0271/740-4050, Büro: H-B 8404

Stand: 14. Juli 2020



---

# Rechnernetze II

SoSe 2020

## 7 Anwendungsprotokolle



### Inhalt

- ➔ Netzwerkmanagement
- ➔ Multimedia-Anwendungen
- ➔ *Overlay-Netze (Peer-to-peer Netze)*
  
- ➔ Peterson, Kap. 9.2.3, 9.3, 9.4
- ➔ Tanenbaum, Kap. 6.4.3, 7.4.5, 5.2.11
- ➔ Kurose/Ross, Kap. 8.1-8.4
- ➔ P. Mahlmann, C. Schindelhauer: Peer-to-Peer-Netzwerke, Springer, 2007.
- ➔ CCNA, Kap. 8

## 7.1 Netzwerkmanagement



### Aufgaben des Netzwerkmanagements

- ➔ Performance-Management
  - ➔ z.B. Überwachung des Verkehrs zur Ressourcenplanung
- ➔ Fehlermanagement
  - ➔ z.B. Ausfall einer Netzwerkkarte, Prüfsummenfehler, *Route-Flapping*, *IP Reassembly*-Fehler
- ➔ Konfigurationsmanagement
- ➔ Accounting-Management
  - ➔ z.B. Nutzungsquoten, Zugriffsrechte auf Ressourcen
- ➔ Sicherheits-Management
  - ➔ Einbruchserkennung, z.B. DoS, Port-Scans

## 7.1 Netzwerkmanagement ...



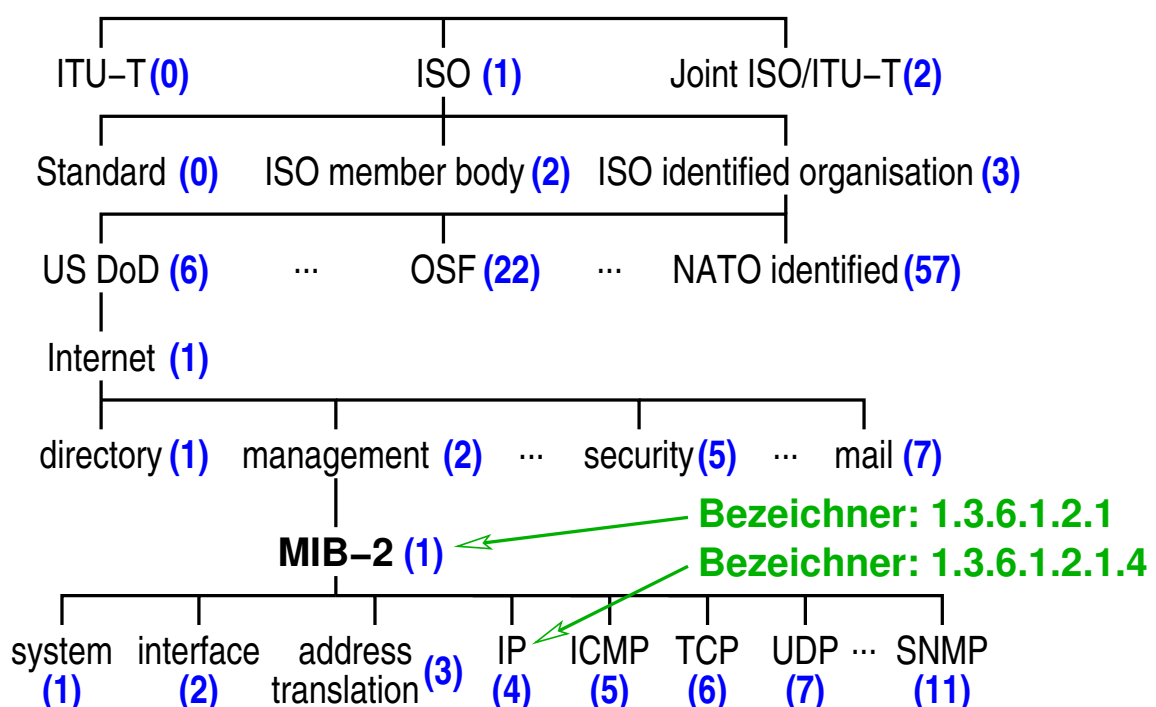
### Simple Network Management Protocol (SNMP) (RFC 1901 ff.)

- ➔ Überwachung des Netzes über das Netz
- ➔ Einfaches Anfrage-Antwort-Protokoll
  - Anfragen i.w.: GET und SET Operationen auf Objekten (Variablen)
- ➔ Setzt bevorzugt auf UDP auf
- ➔ Sicherheitsmechanismen erst ab Version 3 (SNMPv3)
- ➔ Sammlung aller Objekte: *Management Information Base* MIB
  - hierarchisch organisiert (Baumstruktur)
  - beschrieben in der Sprache SMI (Basis: ASN.1)
    - einfache Datentypen, Sequenzen (Tabellen), Strukturen
    - Objekte durch „dezimale Punktnotation“ benannt

## 7.1 Netzwerkmanagement ...



### ASN.1 Objektidentifizierungsbaum





### MIB-Einträge (Beispiele)

- ➔ System: Systemname (HW, BS, ...), Hersteller, *Uptime*, Standort, Kontaktperson, verfügbare Dienste
- ➔ Interface: Hardware-Adresse, gesendete/empfangene Pakete
- ➔ Adreßübersetzung: ARP, insbesondere Übersetzungstabelle
- ➔ IP: Routing-Tabelle, Anzahl Datagramme, Statistiken zu Reassemblierung, verworfene Pakete
- ➔ TCP: Zahl passiver/aktiver *Open*-Operationen, Anzahl *Timeouts*, *Default-Timeout*, Liste offener TCP-Verbindungen
- ➔ UDP: Zahl gesendeter / empfangener / unzustellbarer Datagramme, Liste geöffneter UDP-Ports

## 7.2 Multimedia-Anwendungen



### Im Folgenden:

- ➔ RTP (*Realtime Transport Protocol*) und RTCP (*Realtime Transport Control Protocol*)
  - ➔ Übertragung von z.B. Audio- und Video-Strömen
  - ➔ RTCP: Steuerprotokoll zu RTP
- ➔ SIP (*Session Initiation Protocol*)
  - ➔ Aufbau von Sitzungen (Telefonie, Konferenz)
- ➔ Anmerkung: ITU-Empfehlung H.323 für Internet-Telefonie
  - ➔ umfaßt RTP, RTCP und Protokolle f. Registrierung (H.225), Anrufsignalisierung (Q.931) und Anrufsteuerung (H.245)
  - ➔ SIP ist IETF Alternative zu H.323-Steuerprotokollen



### RTP *Realtime Transport Protocol* (RFC 3550)

- ➔ Allgemeines Transportprotokoll für Multimedia-Ströme
  - unabhängig vom Typ der Daten (Audio, Video, ...) und der Codiermethode (z.B. MPEG)
- ➔ Setzt i.d.R. auf UDP auf
  - typisch: Bibliothek oberhalb der Socket-Schnittstelle
- ➔ Unterstützt das Multiplexen mehrerer Ströme in einen Strom von UDP-Paketen
  - um Bandbreitenbedarf zu verringern
- ➔ Zwei Protokolle: RTP (Daten) und RTCP (Steuerung)
  - RTP: gerade Port-Adresse
  - RTCP: unmittelbar folgende Port-Adresse

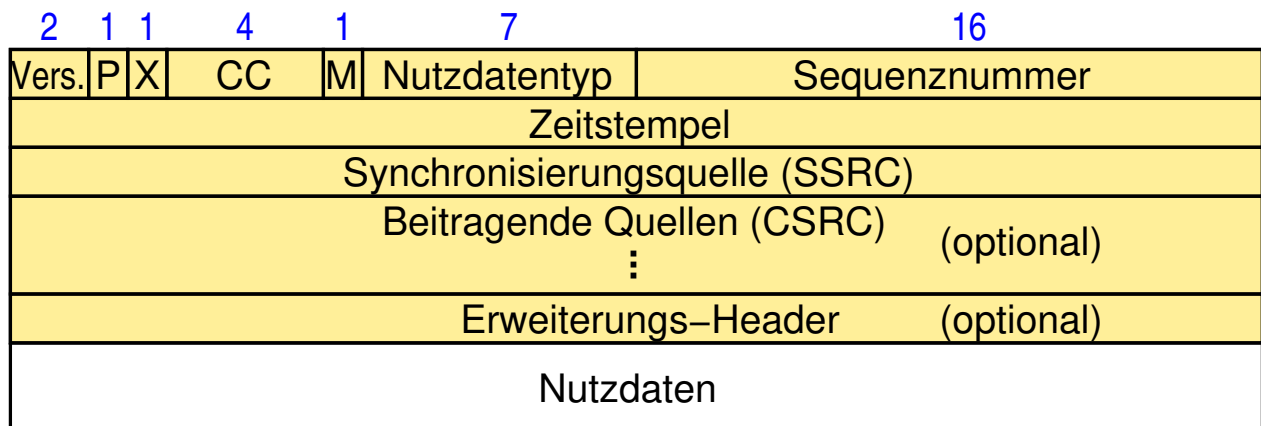


### RTP/RTCP: Anforderungen

- ➔ Pakete mit Zeitstempel (für korrekte Wiedergabe)
- ➔ Synchronisation verschiedener Ströme (z.B. Audio und Video)
- ➔ Empfänger soll Paketverlust erkennen können
  - keine Neuübertragung; Behandlung anwendungsspezifisch
- ➔ *Feedback* zum Sender für Überlastvermeidung
  - Statistiken über Paketverlust; Behandlung anwendungsspezifisch, z.B. aggressivere Kompression
- ➔ Kennzeichnung von Paketen (z.B. Frame-Grenzen bei Video)
- ➔ Geringer Paket-Overhead durch Header
  - i.d.R. kurze Pakete wegen Latenzzeit



### Aufbau eines RTP-Pakets



Vers.: Versionsnummer (2)

P: Füllbit: Nutzdaten aufgefüllt, letztes Byte enthält Anzahl der Füllbytes

X: Erweiterungs-Header vorhanden

CC: Anzahl beitragender Quellen

M: Markierungsbit



### Details zu RTP

- ➔ RTP definiert für jede Anwendungsklasse ein Profil und ein oder mehrere Formate für die Nutzdaten
  - ➔ Profil: z.B. Audio-, Video-Strom; Format: z.B. MPEG, H.261
  - ➔ legt fest, wie Marker / Zeitstempel zu interpretieren sind
- ➔ SSRC ist zufällig gewählte ID der Quelle (z.B. Kamera), die den RTP-Strom identifiziert
- ➔ CSRC nur benutzt, wenn mehrere Stöme durch Multiplex zusammengefaßt werden
  - ➔ SSRC identifiziert dann den Multiplexer
- ➔ Zeitstempel ist relative Zeit
  - ➔ Auflösung durch Profil definiert, je nach Abtastintervall



### Steuerprotokoll RTCP

- ➔ Aufgaben:
  - Rückmeldung über Leistung der Anwendung / des Netzes
  - Synchronisation mehrerer RTP-Ströme eines Senders
  - anwenderfreundliche Senderidentifikation
- ➔ Pakettypen:
  - Empfängerberichte: Statistiken zu Zahl der Pakete, Paketverlust, geschätzter Jitter, ...
  - Senderberichte: Uhrzeit und zugehöriger RTP-Zeitstempel, Zahl bisher gesendete Pakete / Bytes
  - Quellenbeschreibungen: SSRC und kanonischer Name (user@host)

## 7.2.2 Sitzungs- und Anrufsteuerung



- ➔ Aufgaben u.a.:
  - Bereitstellung von Informationen zu einer Konferenz (SDP: *Session Description Protocol*)
    - Name und Zweck der Sitzung
    - Anfangs- und Endezeit
    - Medientypen (Audio / Video)
    - Netzwerk-bezogene Informationen
      - Multicast Adresse, Transportprotokoll (z.B. RTP + Profil, UDP), Ports, Kodierung
  - Verbindungsaufnahme bei Internet-Telefonie (SIP: *Session Initiation Protocol*)
- ➔ (Hier nur Protokolle der IETF betrachtet)

### SDP: *Session Description Protocol* (RFC 4566)

- ➔ Legt i.w. fest, wie Sitzungsinformationen codiert werden
  - ➔ ASCII-Codierung mit *Typ=Wert* Format

```
v=0      SDP-Version
o=larry 2890844526 2890842807 IN IP4 10.0.1.5  Ursprung
s=Networking 101      Sitzungsname
i=A class on computer networking  Sitzungsbeschreibung
u=http://www.cs.princeton.edu  URI der Sitzung
e=larry@princeton.edu  Email-Adresse Kontaktperson
c=IN IP4 224.2.17.12/127  IP-Multicast-Adresse der Sitzung
t=2873397496 2873404696  Start- und Endzeit
m=audio 49170 RTP/AVP 0  Medien (Typ, Port, Transportprotokoll, Format)
m=video 51372 RTP/AVP 31
m=application 32416 udp wp
```

#### Anmerkungen zu Folie 257:

- ➔ Zu o:
  - ➔ Die Felder sind Benutzername, Sitzungs-ID, Sitzungs-Versions-ID, Netzwerk-Typ, Adreß-Typ, Adresse
- ➔ Zu c:
  - ➔ Die 127 nach dem Schrägstrich ist die TTL
- ➔ Zu m:
  - ➔ AVP: Audio/Video Profile
  - ➔ Die Format-Nummern geben den RTP Payload Type an, z.B. 0 = PCMU (Audio 8bit/Sample), 31 = H261 (Video)
  - ➔ Medien-Typ und Medien-Format zusammen entsprechen einem MIME-Type (z.B. application/wp)



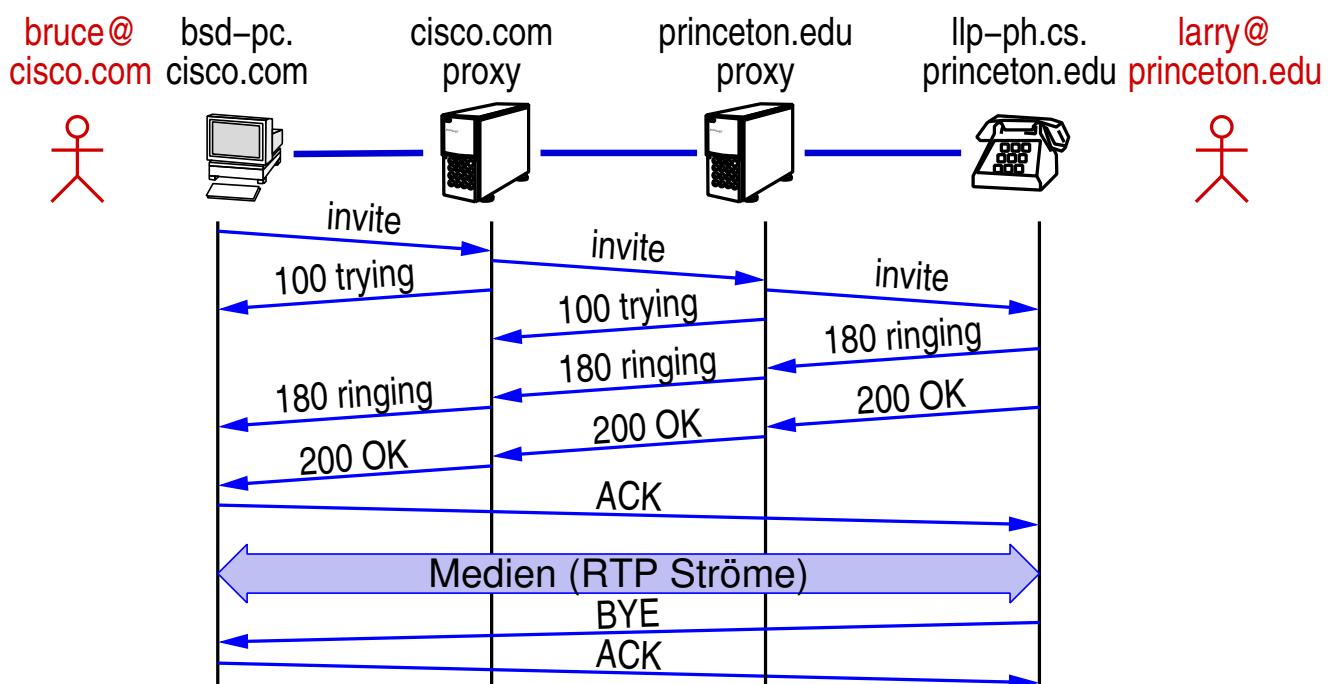


### SIP (*Session Initiation Protocol*) (RFC 3261)

- ➔ Anfrage/Antwort-Protokoll, ähnlich HTTP und SMTP
  - Verwendung von MIME für Nutzdaten
- ➔ Funktionen:
  - Bestimmung des Benutzerstandorts
    - Benutzer kann mehrere verschiedene Rechner nutzen
    - daher: Einführung eines Proxy-Servers, bei dem sich Benutzer mit aktuellem Standort (= Rechner) registriert
  - Feststellung, ob Benutzer an Sitzung teilnehmen kann/will
  - Aushandlung der Medien und Kodierungen
  - Einrichten von Sitzungsparametern (z.B. Port-Nummern)
  - Sitzungsmanagement, z.B. Rufweiterleitung, Änderung von Sitzungsparametern



### Nachrichtenfluß einer SIP-Sitzung



## Anmerkungen zu Folie 259:

In der *invite*-Nachricht stehen die vorgeschlagenen Sitzungsparameter, typischerweise als SDP-Nachricht.

In der *OK*-Nachricht sendet der Angerufene die akzeptierten Sitzungsparameter zurück.

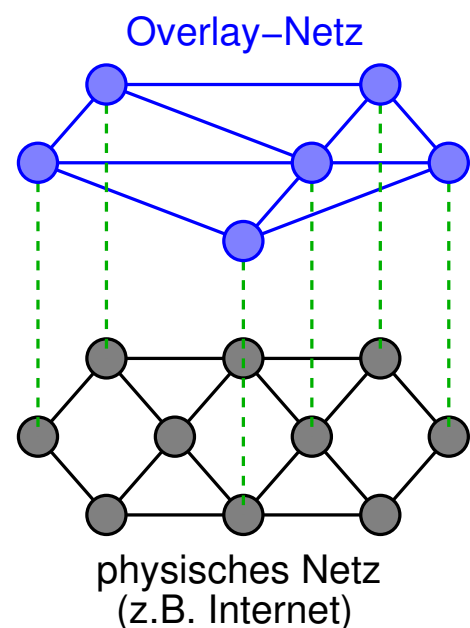
259-1

## 7.3 Overlay-Netze



### Overlay-Netz

- ➔ Logisches Netz, oberhalb eines physischen Netzes implementiert
- ➔ Verbindungen des *Overlay*-Netzes durch Tunnel realisiert
- ➔ Anwendungen:
  - *Routing-Overlays*
    - z.B.: Mbone, 6Bone
  - *Peer-to-Peer-Netze*
    - z.B.: Gnutella, Pastry, (Napster)





### Peer-to-Peer-Netze

- ➔ Dezentral (vollständig verteilt), selbstorganisierend
- ➔ Einsatz: verteilte Speicherung von Daten
  - vollständig verteilte Suche
  - keine zentrale Instanz
- ➔ Vorteile:
  - Skalierbarkeit
  - Ausfallsicherheit
  - (kein Verantwortlicher)



### Peer-to-Peer-Netze: Beispiel Gnutella

- ➔ Knoten des Netzes speichern Datenobjekte
- ➔ Suche von Objekten (QUERY) durch Fluten des Netzes
  - TTL zur Vermeidung unendlicher Zyklen
  - Knoten speichern Zuordnung (QueryID, Sender) aller kürzlich empfangener Anfragen
    - Vermeidung von Zyklen
    - Antwort (QUERY-RESPONSE) kann auf Weg der Anfrage zurückgeschickt werden
  - Antwort enthält IP-Adresse und Port der Rechners, der die Daten besitzt
- ➔ Jeder Knoten kennt mindestens einen weiteren Knoten
  - Kenntnis weiterer Knoten durch Inspizieren von Antworten



### Netzwerkmanagement

- ➔ Performance, Fehler, Konfiguration, Accounting, Sicherheit
- ➔ SNMP: Zugriff auf Objekte (Variablen) über das Netz
- ➔ MIB: Sammlung aller Objekte, hierarchisch strukturiert
  - ➔ System, Interface, ARP, IP, ICMP, TCP, UDP, ...



### Multimedia-Protokolle

- ➔ H.323 Empfehlung: RTP/RTCP und Steuerprotokolle
- ➔ RTP: allgemeines Transportprotokoll, über UDP
  - ➔ Multiplexing von Datenströmen
  - ➔ Zeitstempel
  - ➔ Sequenznummern
  - ➔ Markierung von Paketen
- ➔ RTCP: Steuerprotokoll zu RTP
  - ➔ Sender- und Empfängerberichte
    - ➔ Paketverlust, Jitter
    - ➔ Zuordnung Zeitstempel ↔ Reale Zeit
  - ➔ Quellenbeschreibungen



### Overlay-Netze

- ➔ *Routing-Overlays*
- ➔ *Peer-To-Peer-Netze*
  - ➔ dezentral, selbstorganisierend
  - ➔ verteilte Speicherung von Daten