



50 Jahre Internet – Internetworking 2019

TCP Ports

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

Um TCP-Verbindung aufbauen und aufrechterhalten zu können, müssen Sender und Empfänger jeweils Endpunkte – **Sockets** – für die Kommunikation einrichten

- Jeder Socket ist ausgestattet mit reserviertem Speicherplatz zum Puffern empfangener oder zu sendender Daten
- Zuordnung der Sockets ist auf beide Kommunikationspartner und Zeitdauer der jeweiligen TCP-Verbindung beschränkt
- Jeder Socket besitzt **Socket-Nummer**, die sich aus
 - IP-Adresse des Rechners und
 - lokal zuordenbarer 16-Bit-Nummer – **Port** – ergibt

TCP Ports: Service Access Points der Transportschicht

Ports sind **Service Access Points** der Transportschicht

- Als Teil der Netzwerk-Adresse stellen Ports die Zuordnung von TCP- bzw. UDP-Verbindungen und Datenpaketen zu den Client- bzw. Server-Anwendungen auf den Endsystemen her
- Zu jeder Verbindung gehören zwei Ports, einer auf Seiten des Clients und einer auf Seiten des Servers
- Gültige Portnummer: 0 bis $65535 = 2^{16}$

■ **Well-Known Ports:**

- Ports mit Nummern 0 – 1.023
- Weltweit eindeutig für Standarddienste reserviert, z.B.
80 – HTTP, 23 – Telnet, 53 – DNS, ...

- Portnummern: 1.024 – 65.535 können beliebigen Anwendungsprogrammen zugeordnet werden

Man unterscheidet weiter:

■ **Reservierte Ports:**

- 0 – 255 für TCP/IP-Anwendungen
- 256 – 1.023 für bestimmte UNIX-Anwendungen

■ **Registrierte Ports:**

- 1.024 – 49.151 von IANA registriert

■ **Private, dynamische Ports:** Rest

TCP Ports

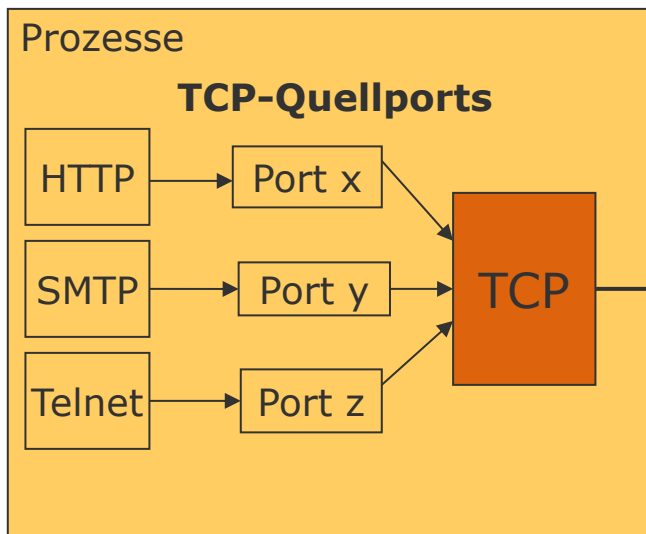
Portnummern werden beim Verbindungsaufbau zwischen zwei Anwendungsprogrammen gewählt:

- **Zielport:** Portnummer der gewünschten Anwendung
- **Quellport:** Nicht reservierte und zur Zeit nicht anders genutzte Portnummer
- TCP-Verbindung kann anhand der beiden Socket-Nummern (Socket Sender, Socket Empfänger) eindeutig identifiziert werden
- Server-Socket kann gleichzeitig mehreren Verbindungen dienen
- Anwendungsprogramme nutzen TCP-Transportdienst über sogenannte **TCP-Primitive**, die über den Socket implementiert sind, z.B. Request, Response, Confirm, Read, Write, ...

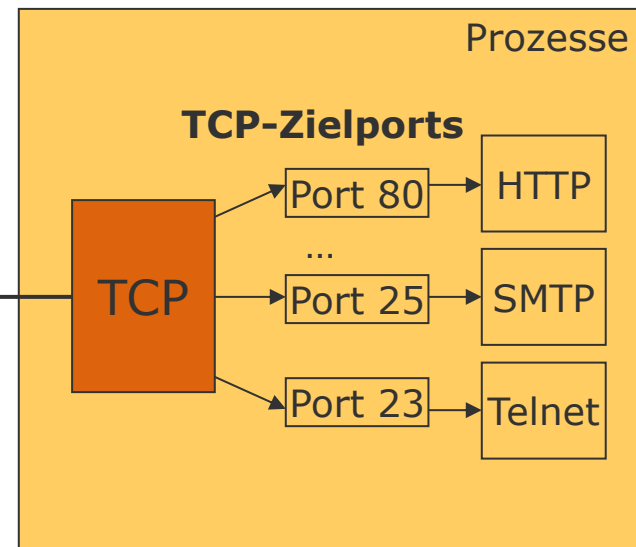
TCP Ports

TCP-Verbindungen zwischen verschiedenen Anwendungsprozessen verlaufen über TCP Ports

Rechner A



Rechner B



TCP - Transport