

Kommunikationsarten

Leitungs- vs. paketbasierte Verbindungen

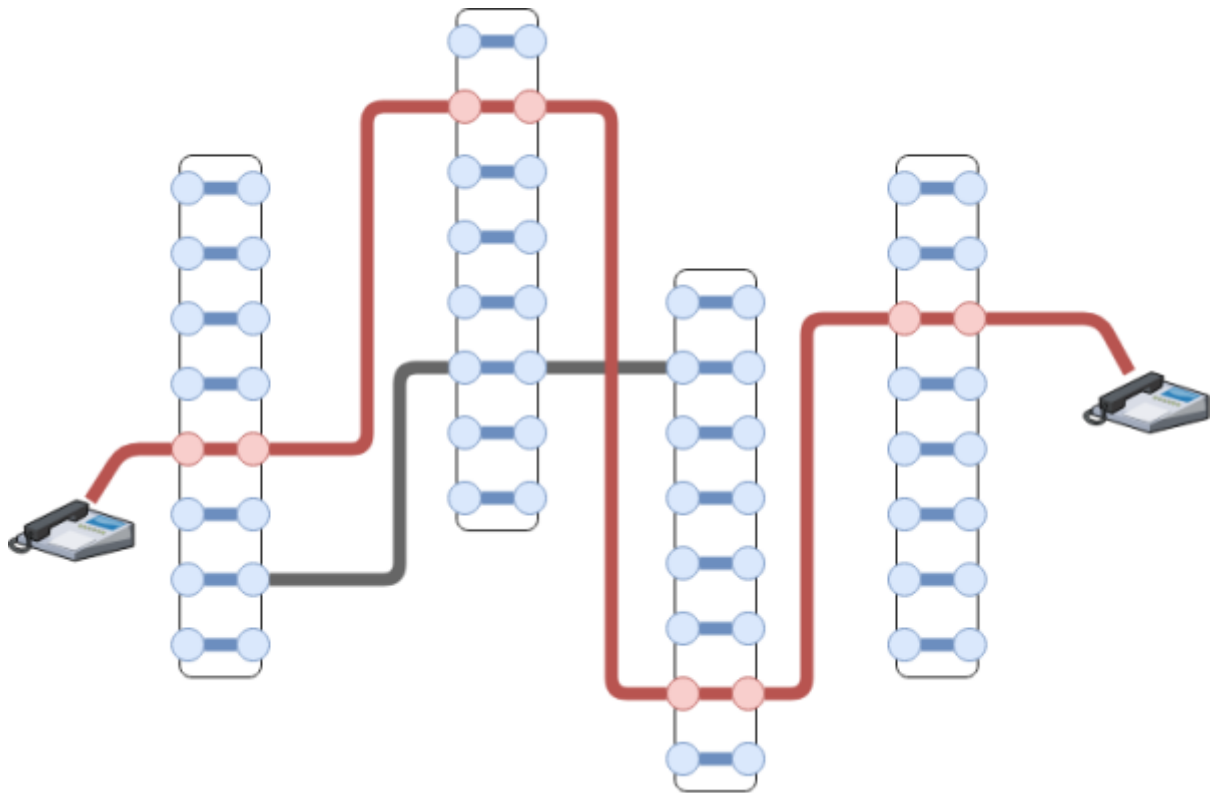
Leitungsbasierende Kommunikation

Leitungsvermittelte Verbindungen öffnen einen exklusiven Kommunikationskanal zwischen zwei Kommunikationspartnern - ein Beispiel ist das frühere Telefonnetz.



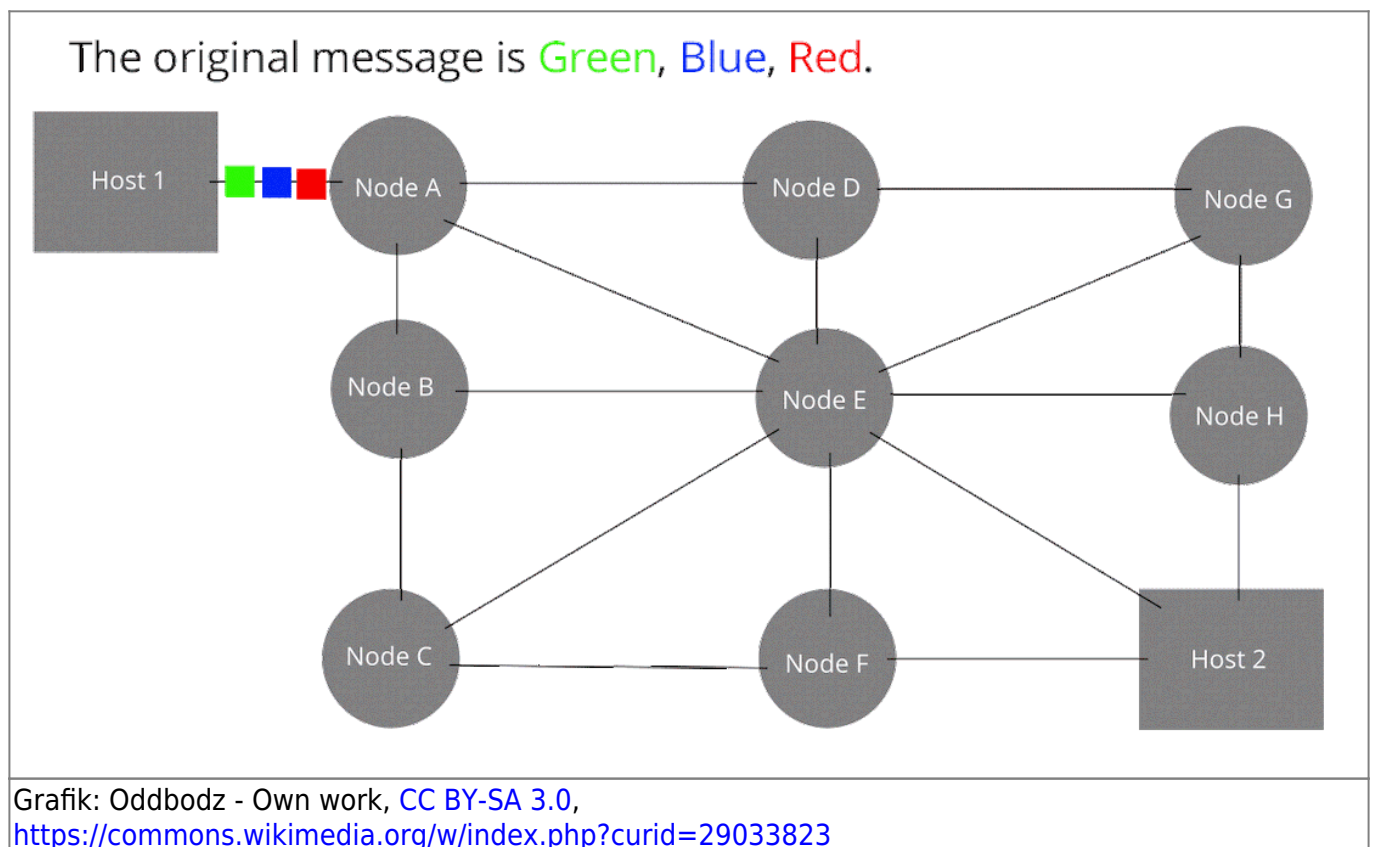
1)

Durch Änderung der Steckverbindung, später durch Relais und elektronische Schaltungen wurden die Teilnehmer direkt verbunden. Der gesamte Kommunikationsvorgang nutzte diesen Kanal. Bei leitungsbasierten Verbindungen ist der logistische Aufwand recht hoch, die Vermittlungszeit ebenfalls, dafür kann man sehr genau vorhersagen, mit welcher Bandbreite die Kommunikation durchgeführt werden kann, da die Verbindung für diesen einen Kommunikationsvorgang "reserviert" ist.



Paketbasierte Kommunikation

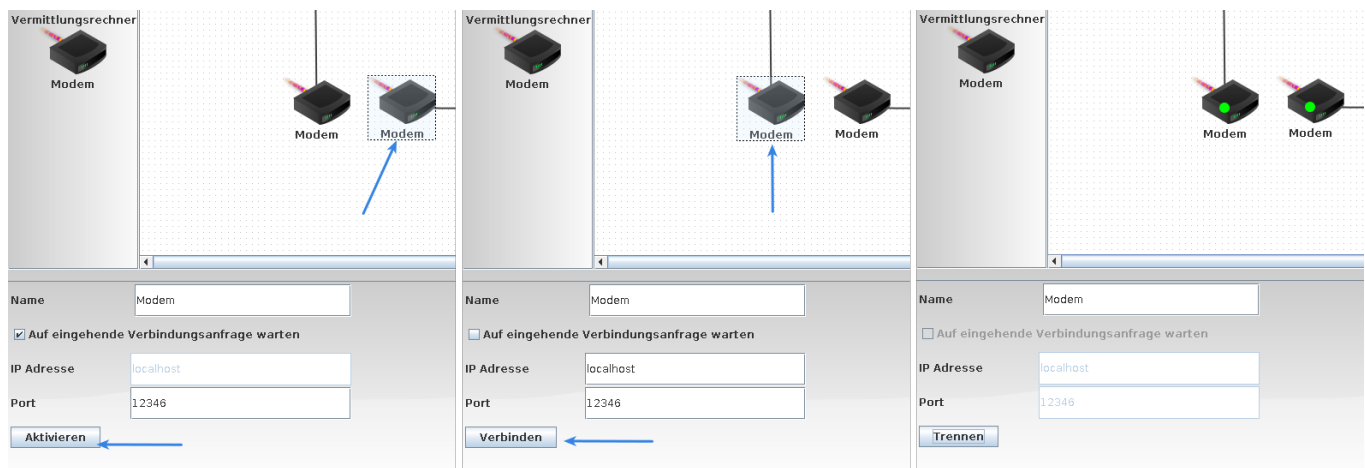
Bei der Paketbasierten Kommunikation (z.B. per TCP oder UDP), werden die Kommunikationsinhalte in "Datenpakete aufgeteilt und über ein vermaschtes Netz an den Empfänger vermittelt - jedes Paket einzeln. Dabei können verschiedene Teile der Kommunikation durchaus unterschiedliche Wege durch das Netzwerk nehmen und in anderer Reihenfolge am Ziel ankommen, als sie abgesendet wurden.



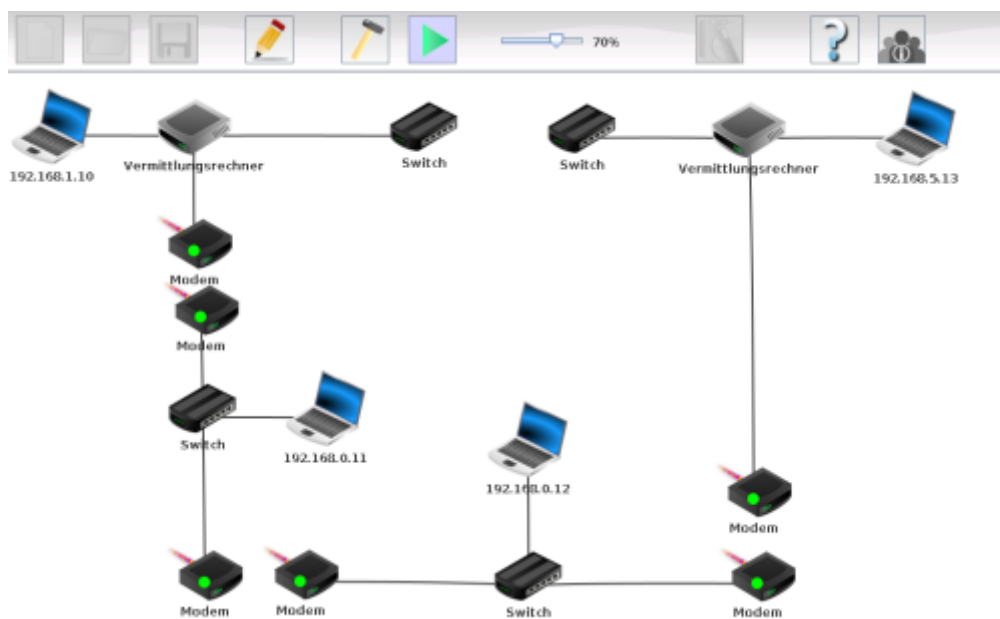
Ein Experiment mit Filius

Öffne das folgende Filius-Szenario: [paketbasierte_kommunikation.flx](#) und aktiviere zunächst **alle** Modemverbindungen.

Beim empfangenden Modem (das auf eingehende Verbindungsanfragen wartet) die Schaltfläche "Aktivieren klicken", dann beim Gegenstück auf "Verbinden". Wenn die Verbindung zustande gekommen ist, haben beide Modems einen grünen Punkt:



Reduziere dann die Ausführungsgeschwindigkeit auf ca. 70%. Das Ergebnis sollte so aussehen:



(A1)

- Öffne auf dem Laptop mit der IP-Adresse 192.168.1.10 (links oben) eine Kommandozeile
- Pinge den Rechner mit der Adresse 192.168.0.11 an und beobachte am Blinken der

Verbindungsleitungen, welchen Weg die ICMP Pakete nehmen.

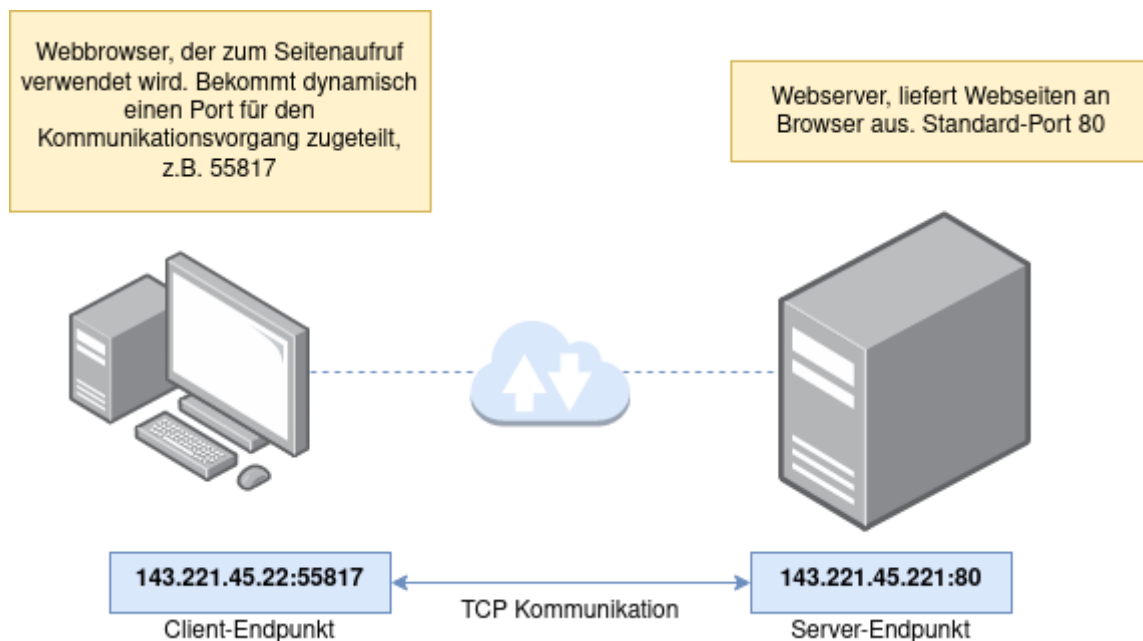
- Wiederhole den Versuch mit dem Zielrechner 192.168.0.12. Was fällt dir auf? Kannst du anhand der Konfiguration der Rechner (mit den Mauszeiger über dem Rechner stehen bleiben) eine Erklärung für das Verhalten finden?
- Füge zwischen den beiden Switches oben ein Kabel ein.
- Wiederhole die beiden Versuche von oben und notiere deine Beobachtungen. Erläutere den Begriff "Paketbasierte Kommunikation" anhand deiner Beobachtungen.

TCP genauer angeschaut

Mit dem TCP Protokoll kann man Informationen in beide Richtungen übertragen. Kommunikation über TCP ermöglicht es, zur gleichen Zeit Daten zu senden und zu empfangen. Die Übertragungseinheiten, auf die TCP bei der Datenübertragung zurückgreift, sind Segmente (Pakete), die zusätzlich zu den Nutzdaten auch Metainformationen enthalten können. Die maximale Paketgröße ist auf 1.500 Byte beschränkt (MTU - "Maximum Transfer Unit"). Verbindungsauf- und -abbau und auch die Datenübertragung werden von TCP-Software im Netzwerk-Protokollstapel des Betriebssystems übernommen.

Der TCP-Stack des Betriebssystems wird von den Netzwerkanwendungen wie Webbrowsern oder Serverprogrammen über spezifische Schnittstellen bedient, jede Verbindung ist dabei durch zwei Endpunkte - **Client** und **Server** - definiert. Welcher Kommunikationspartner die Client- und welche die Serverrolle übernimmt, spielt dabei zunächst keine Rolle - wichtig ist, dass der TCP-Stack an jedem Endpunkt ein eindeutiges, Paar aus **IP-Adresse** und **Port** verwenden kann. Ein solches Paar nennt man auch einen **Socket**.

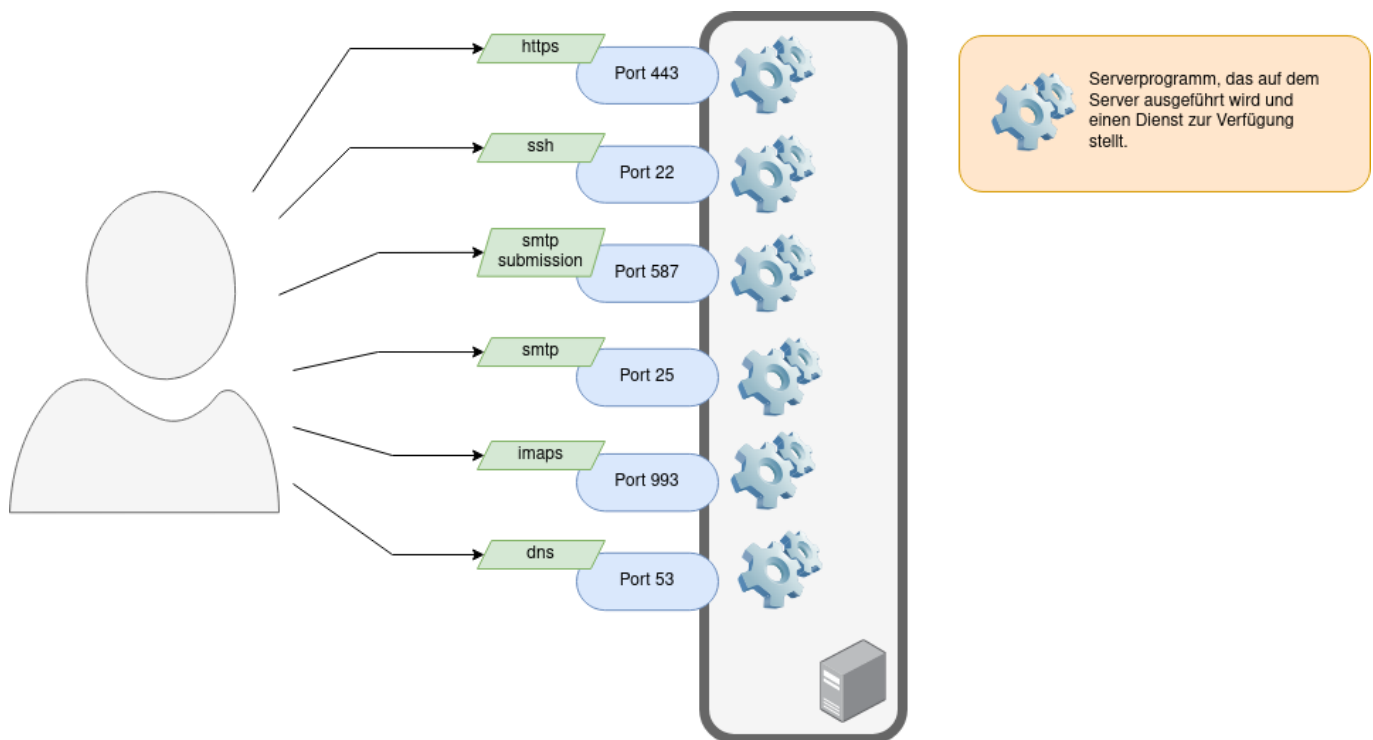
Beispiel:



Exkurs: Sockets

Ein **Server** ist ein Programm, das Dienste zur Verfügung stellt, häufig wird es auf einem Server (Gerät) dauerhaft im Hintergrund ausgeführt. Es stellt seine Dienste an einem Port zur Verfügung.

Zusammen mit der IP-Adresse des Servergeräts kommuniziert es also über einen Socket mit der Außenwelt. So kann ein Servergerät viele Serverdienste anbieten, ohne dass es zu Kollisionen kommt.



Ein Client ist ein Programm, das auf die Dienste eines Serverprogramms zugreift. Es wird meist auf dem Clientgerät ausgeführt, z.B. dem Laptop oder Smartphone, mit dem ein Benutzer mit Netzwerkdiensten in Interaktion tritt.

Für jede aktive Netzwerkverbindung verwendet auch das Clientgerät einen eigenen Socket:

lsuf -i

COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE	NAME
mattermos	62969	frank	24u	IPv4	14123862	0t0	TCP	pike.local:42066->chat.schule.social:https (ESTABLISHED)
mattermos	62969	frank	26u	IPv4	14286251	0t0	TCP	pike.local:42076->chat.schule.social:https (ESTABLISHED)
GeckoMain	63117	frank	127u	IPv4	14273063	0t0	TCP	pike.local:37696->codeberg.org:https (ESTABLISHED)
GeckoMain	63117	frank	128u	IPv4	14293606	0t0	TCP	pike.local:51834->d.schule.social:https (ESTABLISHED)
GeckoMain	63117	frank	145u	IPv4	14271497	0t0	TCP	pike.local:41772->ec2-52-38-31-225.us-west-2.compute.amazonaws.com:https (ESTABLISHED)
electron	65722	frank	26u	IPv4	14301860	0t0	TCP	pike.local:51838->d.schule.social:https (ESTABLISHED)
thunderbi	393015	frank	66u	IPv4	14297360	0t0	TCP	pike.local:39816->pd95563b6.dip0.t-ipconnect.de:imaps (ESTABLISHED)
thunderbi	393015	frank	75u	IPv4	14121927	0t0	TCP	pike.local:51426->mbox1.belwue.de:imaps (ESTABLISHED)
thunderbi	393015	frank	77u	IPv4	14301460	0t0	TCP	pike.local:44966->185.67.36.168:imap (ESTABLISHED)
thunderbi	393015	frank	100u	IPv4	14297324	0t0	TCP	pike.local:39810->pd95563b6.dip0.t-ipconnect.de:imaps (ESTABLISHED)
thunderbi	393015	frank	103u	IPv4	14301601	0t0	TCP	pike.local:39818->pd95563b6.dip0.t-ipconnect.de:imaps (ESTABLISHED)
thunderbi	393015	frank	105u	IPv4	14294903	0t0	TCP	pike.local:52434->mail.lehrerpost.de:imaps (ESTABLISHED)
thunderbi	393015	frank	108u	IPv4	14294908	0t0	TCP	pike.local:48992->mbox1.belwue.de:imap (ESTABLISHED)
thunderbi	393015	frank	132u	IPv4	14295665	0t0	TCP	pike.local:51448->mbox1.belwue.de:imaps (ESTABLISHED)
thunderbi	393015	frank	133u	IPv4	14294922	0t0	TCP	pike.local:44964->185.67.36.168:imap (ESTABLISHED)
ssh	2736000	frank	3u	IPv4	14122494	0t0	TCP	pike.local:60050->helga.cartoons.home:ssh (ESTABLISHED)
ssh	2736013	frank	3u	IPv4	14124285	0t0	TCP	pike.local:47042->snorre.cartoons.home:ssh (ESTABLISHED)

Lokaler Socket
der Verbindung

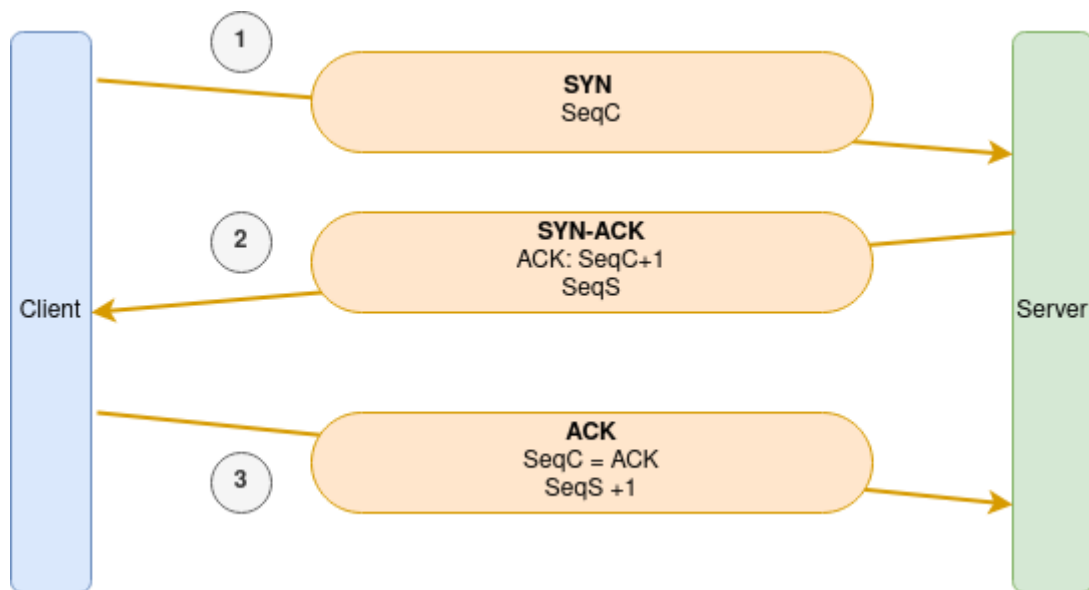
Entfernter Socket. Portnummern
durch Funktion ersetzt
(993 → imaps...)

TCP-Verbindungsaufbau: Three-Way-Handshake

Damit Client und Server eine TCP Verbindung aufbauen können, müssen beide über eine IP-Adresse verfügen und den Port für die Datenübertragung definiert und freigegeben haben²⁾.

Der Ablauf beim Verbindungsaufbau mit TCP sieht folgendermaßen aus:

- Im ersten Schritt sendet der Client dem Server ein **SYN-Paket** (**s**ynchronize) mit einer individuellen, zufälligen Sequenznummer. Diese Nummer stellt die Übertragung in der korrekten Reihenfolge und ohne Duplikate sicher.
- Hat der Server das Paket erhalten, stimmt er dem Verbindungsaufbau zu, indem er ein **SYN-ACK-Paket** (**a**cknowledgement) zusammen mit der um 1 erhöhten Sequenznummer des Clients zurückschickt. Außerdem übermittelt er dem Client eine eigene (zufällige) Sequenznummer.
- Zum Abschluss des Verbindungsaufbaus bestätigt der Client den Erhalt des SYN-ACK-Segments, indem er ein eigenes ACK-Paket versendet, das die um 1 erhöhte Sequenznummer des Servers enthält.



(A2)

Öffne in Filius das folgende Beispiel [webserver_client.flr](#). Auf dem Laptop ist ein Webbrowser installiert, öffne mit diesem die Webseite auf den Server (192.168.0.1)

Vollziehe den TCP Verbindungsaufbau nach.

Kannst in der Analyse des Datenverkehrs herausfinden, ob und wie eine TCP Verbindung auch beendet wird? Recherchiere und erstelle einen Heftaufschrieb.

Material

01_kommunikationsarten.odp	1.3 MiB	15.12.2021	18:59
01_kommunikationsarten.pdf	721.9 KiB	15.12.2021	18:59
02_besprechung_a2.odp	53.4 KiB	15.12.2021	16:49
02_besprechung_a2.pdf	59.4 KiB	15.12.2021	16:49

3wayhs.drawio.png	22.6 KiB 15.12.2021 16:37
3wayhs.png	28.7 KiB 05.10.2020 18:07
auswahl_109.png	67.7 KiB 15.12.2021 15:56
auswahl_110.png	79.2 KiB 15.12.2021 15:57
filius_modem.png	50.9 KiB 15.12.2021 15:55
leitungsbasiert.png	34.5 KiB 05.10.2020 17:07
lsof.png	414.4 KiB 15.12.2021 19:05
packet_switching.gif	7.5 MiB 05.10.2020 17:14
server-client.png	34.9 KiB 05.10.2020 17:50
serverdienste.drawio.png	82.6 KiB 15.12.2021 19:00
vermittlung.jpg	70.9 KiB 05.10.2020 16:52

1)

Bildquelle: <https://commons.wikimedia.org/wiki/File:Telefonistinde.jpg>, Lizenz: [Creative-Commons-Lizenz "Namensnennung 3.0 nicht portiert"](#), Beschreibung: *Dansk: Manuel betjening på telefoncentral*
Datum: 2. April 2017 Quelle: *MyNewsDesk.com* Urheber: *Arkiv*

2)

Hier greifen z.B. Firewalls ein, indem Sie den Zugriff auf bestimmte Ports blockieren

From:
<https://info-bw.de/> -

Permanent link:
https://info-bw.de/faecher:informatik:oberstufe:netzwerke:kommunikationsarten_und_tcp:start?rev=1639595139

Last update: **15.12.2021 19:05**

