Zusammenfassung M129

LAN-Komponenten in Betrieb nehmen

2008-11-12

Emanuel Duss

Über

Autor	Emanuel Duss
Erstellt	2008-06-11
Bearbeitet	2008-11-12
Heute	2008-11-12
Bearbeitungszeit	07:25:04
Lehrjahr	1. Lehjahr 2006/2007
Pfad	/home/emanuel/Daten/Lehre/2_Lehrjahr/129_LAN-Komponenten_in_Betrieb_nehmen/M129_Zusammenfassung.odt
CC-Lizenz	 Creative Commons Namensnennung-Keine kommerzielle Nutzung-Weitergabe unter gleichen Bedingungen 2.5 Schweiz http://creativecommons.org/licenses/by-nc-sa/2.5/ch/
Powered by	  

Bearbeitungsprotokoll

Datum	Änderung(en)
2008-06-11	Erstellt
2008-09-03	PDF für Teilprüfung erstellt

Inhaltsverzeichnis

1 Netzwerk-Modelle

1.1 ISO/OSI-Modell

Layer	Name	Inhalt des Layers
7	Application Layer	
6	Presentation Layer	
5	Session Layer	
4	Transport Layer	
3	Network Layer	Datagramme; Subnetting, DHCP, Routertabellen,
2	Data Link Layer	Ethernet-Frames; MAC-Sublayer, LLC-Sublayer, MAC-Adressen, ARP, Spanning-Tree-Protocol
1	Physical Layer	Autonegotation, Halb- Vollduplex, Leitungscodes, Topologie

1.2 TCP-IP-Modell

1.2.1 Fünfschichtig

Schicht	**Protokolle**	**Adressierungstyp**	**ISO-OSI**
Application	HTTP, FTP, Telnet, SMTP, POP,		5, 6, 7
Transport	TCP / UDP	Port	4
Internet	IPv4, IPv6	IP	3
Data-Link	Ethernet, Token Bus, Token Ring	MAC#	2
Physical			1

1.2.2 Vierschichtig

TCP/IP-Schicht	**≈ OSI-Schicht**	**Beispiel**
Anwendungsschicht	5–7	HTTP, FTP, SMTP, POP, Telnet
Transportschicht	4	TCP, UDP
Vermittlungsschicht	3	IPv4,IPv6
Netzzugangsschicht	1–2	Ethernet, Token Bus, Token Ring, FDDI

ICMP: Internet Control Message Protocol. Wird beim Ping verwendet.

1.2.3 Beschreibungen der einzelnen Schichten[1]

- **Anwendungsschicht** (application layer): Die Anwendungsschicht umfasst alle Protokolle, die mit Anwendungsprogrammen zusammenarbeiten und die Netzwerkinfrastruktur für den Austausch anwendungsspezifischer Daten nutzen.

- **Transportschicht** (transport layer): Die Transportschicht stellt eine Ende-zu-Ende-Verbindung her. Das wichtigste Protokoll dieser Schicht ist das Transmission Control Protocol (TCP), das Verbindungen zwischen jeweils zwei Netzwerkteilnehmern zum zuverlässigen (nicht "sicheren", da das Wort "sicher" im Sinne von fälschungssicher/abhörsicher gebraucht wird) Versenden von Datenströmen herstellt. Es gehören aber auch Datagramm-Protokolle – zum Beispiel das User Datagram Protocol (UDP) – in diese Schicht, bei denen nur die Zustellung an den richtigen Dienst zuverlässig gemacht und keine Verbindung aufgebaut wird.

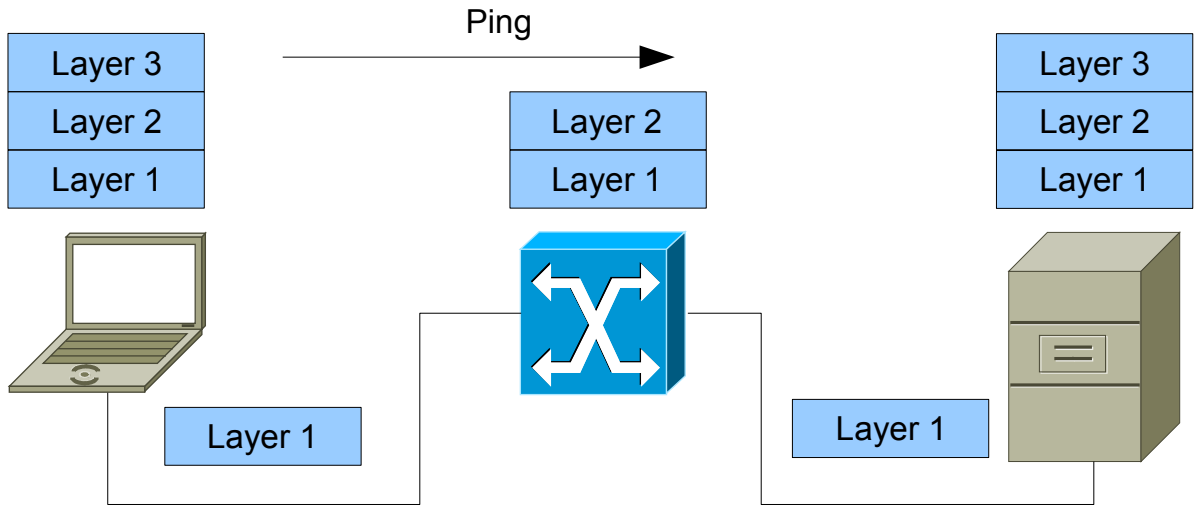
- **Vermittlungsschicht** (internet layer): Die Vermittlungsschicht ist für die Weitervermittlung von Paketen und die Wegewahl (Routing) zuständig. Auf dieser Schicht und den darunterliegenden Schichten werden Punkt-zu-Punkt-Verbindungen betrachtet. Die Aufgabe dieser Schicht ist es, zu einem empfangenen Paket das nächste Zwischenziel zu ermitteln und das Paket dorthin weiterzuleiten. Kern dieser Schicht ist das Internet Protocol (IP), das einen Paketauslieferungsdienst bereitstellt. Die Vermittlungsschicht entspricht im ISO/OSI-Referenzmodell der Vermittlungsschicht.

- **Netzzugangsschicht** (auch: host-to-network layer): Die Netzzugangsschicht ist im TCP/IP-Referenzmodell spezifiziert, enthält jedoch keine Protokolle der TCP/IP-Familie. Sie ist vielmehr als Platzhalter für verschiedene Techniken zur Datenübertragung von Punkt zu Punkt zu verstehen. Die Internet-Protokolle wurden mit dem Ziel entwickelt, verschiedene Subnetze zusammenzuschlieflen. Daher kann die Host-an-Netz-Schicht durch Protokolle wie Ethernet, FDDI, PPP (Punkt-zu-Punkt-Verbindung) oder 802.11 (WLAN) ausgefüllt werden. Die Netzzugangsschicht entspricht im ISO/OSI-Referenzmodell der Sicherungs- und Bitübertragungsschicht.

1 Quelle: Wikipedia

2 Layer 2: Data Link Layer

2.1 Ping über einen Switch



2.2 Aufbau Ethernet II Frame (Ethernet DIX)

Das DIX-Konsortium standardisierte die Darstellung von Ethernet-Frames. Diese nennt man auc Ethernet II Frames.

Destination Address	Source Address	Type	Data / Nutzlast
6 Byte	6 Byte	2 Byte	64 bis 1518 Byte
Ethernet II Header			Data

2.2.1 MAC-Adressen Aufbau

I/G	U/L	Hersteller	Karten-ID
1 Bit	1 Bit	22 Bit	24 Bit
48 Bit = 6 Byte			

- I = 0 = Individuell = Unicast
- G = 1= Gruppe = Multicast oder Broadcast
- U = 0= Universell = von IEEE-OUI Konvention erzeugte Adresse
- L = 1 = Lokale Adresse = nur lokal verwendbar (durch Software erzeugte MAC-Adresse, die sich nicht an die OUI Einteilung der IEEE hält

Unicast	Zustellung an ein Ziel
Multicast	Zustellung an mehrere Ziele (Streaming, Spanning Tree Protocol (STP), IRC)
Broadcast	Zustellung an alle Hosts innerhalb der gleichen Broadcast-Domäne

Unicast: 00:45:A9:A8:F1:69; genau ein Ziel

Broadcast: FF:FF:FF:FF:FF:FF; geht an alle

2.2.2 Ethernet-Type-Field

Zu welchem Layer 3-Protokoll gehört die zu transportierende Nutzlast?

IANA legt diese fest:

Ether-Type	Layer 3 Protokoll
0x0800	IP
0x0806	ARP
0x8100	VLAN tagged Frames

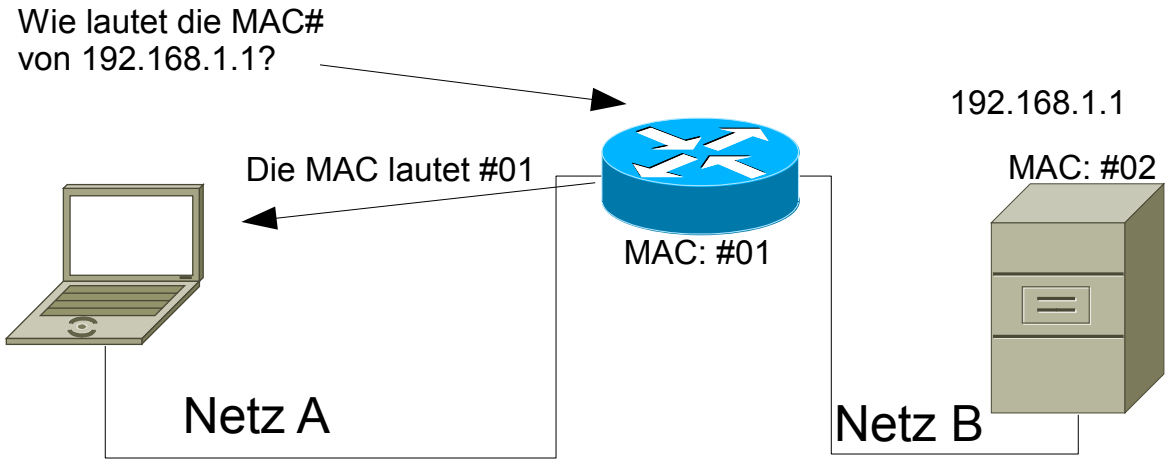
2.2.3 MAC-Adresse rausfinden

Linux: ifconfig

Windows ipconfig /all oder getmac (was jedoch sinnlos ist)

2.3 Spezielle ARP-Nachrichten

2.3.1 Proxy ARP[2]



Proxy ARP erlaubt einem Router, ARP-Anforderungen für Hosts zu beantworten.

Die Hosts befinden sich dabei in verschiedenen Netzen. Bei der Kommunikation ist für die Hosts der Router transparent, das heiflt er braucht nicht speziell angesprochen zu werden, sondern die Hosts können wie gewöhnlich Pakete über verschiedene Netze hinweg versenden.
Sendet Computer A eine ARP-Anforderung an Computer B, reagiert der dazwischen liegende Router anstelle des Computers B mit einer ARP-Antwort und der

2 Quelle: Wikipedia

Hardwareadresse der Schnittstelle (MAC des Ports am Router), auf der die Anfrage empfangen wurde. Der anfragende Computer A sendet dann seine Daten an den Router, der sie dann an Computer B weiterleitet.
Proxy ARP kann man am ARP-Cache von Computer A erkennen. Falls für mehrere IP-Adressen dieselbe MAC-Adresse eingetragen ist, arbeitet der Router mit dieser MAC-Adresse als Proxy. Die Einträge können auch ein Hinweis auf einen Angriff durch ARP-Spoofing sein.

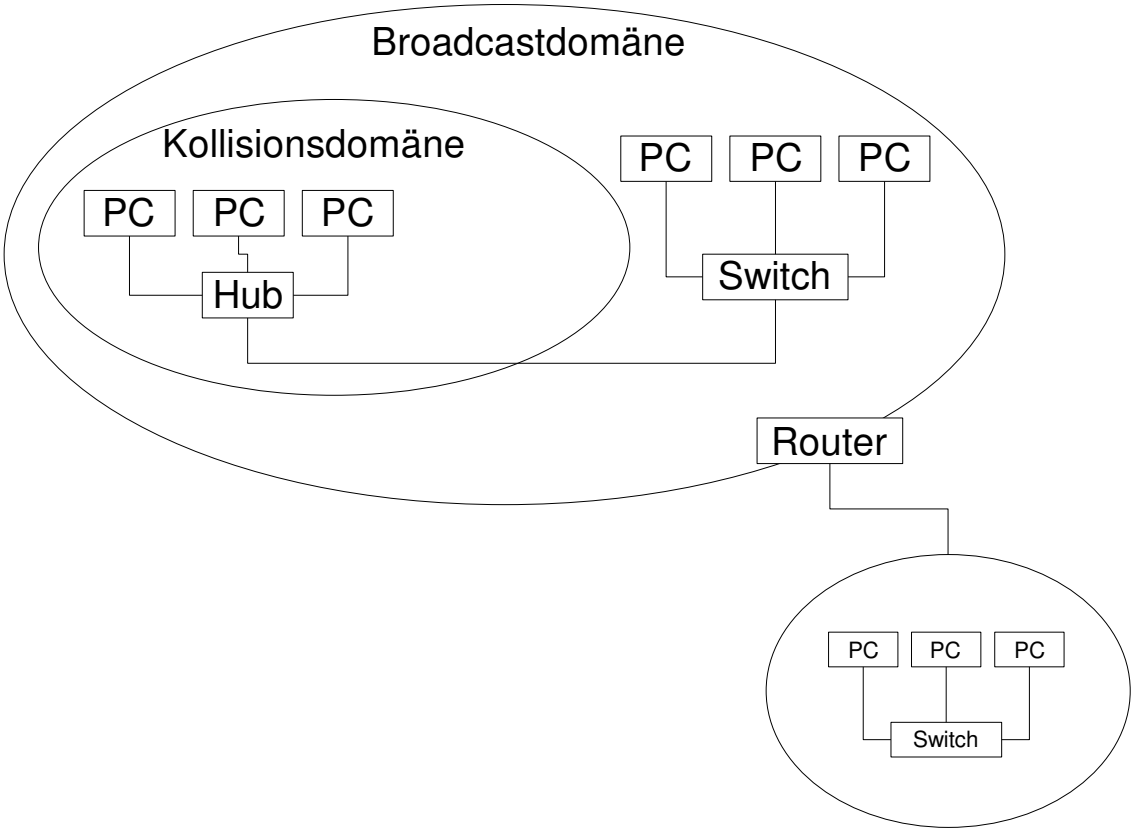
2.3.2 Gratious ARP[3]

Gratuitous ARP (engl. „unaufgefordertes ARP“) bezeichnet eine spezielle Verwendung von ARP. Dabei sendet ein Host ein ARP-Anforderungs-Broadcast, bei dem er seine eigene IP-Adresse als Quell- und Ziel-IP-Adresse einträgt. Damit teilt er seine ggf. neue MAC-Adresse unaufgefordert mit. Das kann mehreren Zwecken dienen:

1. Normalerweise darf keine Antwort kommen, denn eine IP-Adresse muss in einem Netz eindeutig sein. Bekommt er trotzdem eine Antwort, ist das für den Administrator ein Hinweis darauf, dass ein Host nicht richtig konfiguriert ist.
2. Jeder Host aktualisiert seinen ARP-Cache. Das ist beispielsweise dann nützlich, wenn die Netzwerkkarte eines Rechners ausgetauscht wurde und die anderen Hosts über die neue MAC-Adresse informiert werden sollen. Gratuitous ARP geschieht deshalb normalerweise beim Booten eines Computers.
3. Wenn zwei Server aus Gründen der Ausfallsicherheit als Server und Ersatzserver aufgebaut sind und sich eine IP-Adresse teilen und der aktive Verkehr vom einen auf den anderen geschwenkt werden soll, ist die IP-Adresse jetzt über eine andere MAC-Adresse zu erreichen. Diese neue MAC-/IP-Adress-Zuordnung muss bekannt gemacht werden. Sonst bekommt niemand den Wechsel mit.
4. In einem Mobile IP-Szenario sendet der Home Agent einen Gratuitous ARP, wenn sich der Mobile Host aus dem Heimatnetz entfernt, um die Pakete stellvertretend für diesen zu empfangen. Analog sendet der Mobile Host einen Gratuitous ARP, sobald er sich wieder im Netz befindet.

3 Quelle: Wikipedia

2.4 Domänen



2.4.1 Kollisionsdomäne

Teil in einem LAN, indem Ethernet-Frames von Hosts mit anderen Ethernet-Frames kollidieren können. Fehler werden i.d.R. Vom Layer 2A-Protokoll CSMA/CD erkannt.

Die Kollisionsdomäne will man natürlich möglichst klein halten. Dies macht ein Switch, da der die Ethernet-Frames gezielt an den Zielhost weiterleitet. Dann hat man die Kollisionsdomäne auf diese Linkstrecke reduziert.

2.4.2 Broadcast-Domäne

Den Teil in einem LAN, indem sich MAC-Broadcast adressierte Frames ausbreiten können. Dieser Teil besteht aus Layer 2 Netzwerk-Komponenten. Router leiten MAC-Broadcasts nicht mehr weiter. Die MAC-Broadcast-Domäne überspannt somit die gesamte Layer 2 Infrastruktur. Die MAC-Broadcast-Domänen werden mit Routern getrennt.

Man braucht diese Broadcasts für ARP, DHCP, NetBIOS, etc... Diese kann man nicht verhindern.

2.5 Switching

2.5.1 Warum einen Switch einsetzen?

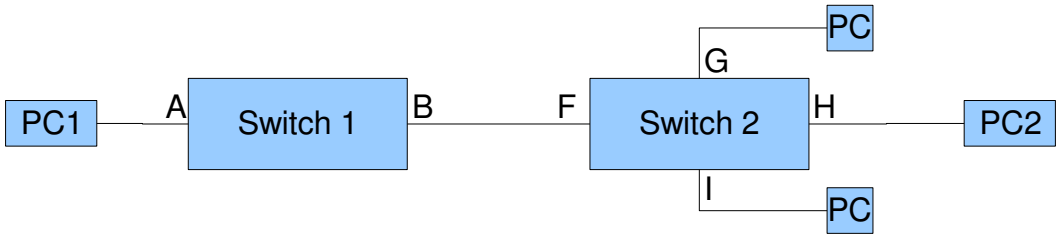
- Der Switch leitet Datenpakete gezielt weiter.
- Die Kollisionsdomäne beschränkt sich auf ein dynamisch entstehendes Mikrosegment (ist also sehr klein).
- Autonegotation: Der Switch handelt automatisch die schnellste Übertragungsrate pro Verbindung aus.
- Full-Duplex-Mode: Der Switch kann gleichzeitig senden und empfangen.
- Der Switch unterstützt die Flusskontrolle
- Schleifenbildung können durch das Spanning Tree Protocol (STP) verhindert werden.
- Bei Layer-3-Switches kann man VLANS einrichten.

2.5.2 Wie ein Switch switcht

1. Switch bekommt einen Input (DIX-Frame mit Destination und Source)
2. Wenn ein Gerät etwas sendet erkennt der Switch in der SRC# die und trägt diese beim Port ein, an dem dieses Frame ankommt.
3. Switch schaut in der Switching-Table oder Forwarding-Table nach, an welchen Internetn Port er das Frame weiterleiten muss:

MAC	Port/Interface
Source	Port 1
Source#	Port n

2.5.3 Was macht der Switch, wenn er nicht weiss, wo die DST ist?



1. PC01 sendet PC02 ein Ethernet-Frame. Nicht vergessen: die SRC# ist in diesem

Frame!

2. Der Switch erhält das Frame. Nun macht er in der Forwarding-Tabelle einen Eitrag, auf welchem Port die SRC-MAC-Adresse liegt.
3. **Der Switch weis jedoch nicht wo die DST# ist. Deshalb schickt er dieses Paket an alle Ports.**
4. Das Paket kommt deshalb auch beim PC02 an. Dieser antwortet auf dieses Paket.
5. Im Antwort-Frame vom PC02 ist die SRC# vom PC02 eingetragen. Nun trägt der Switch auch diese Adresse in der Forwarding-Tabelle ein.

2.5.4 Lernen und Forwarden

Beim Lernen schaut der Switch die Pakete an und schaut dabei auf welchem Port die SCR# kommt. Diese trägt er dann in der Forwarding-Tablle ein.

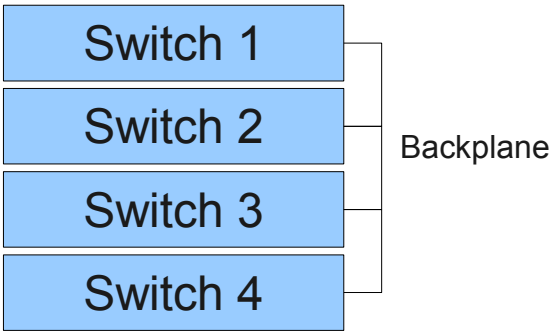
Das Aging gibt an, wie lange eine MAC# in der Forwarding-Table gespeichert wird.

Im Forwarding-Modus leitet er die DST# an den dazu eingeteilten Port weiter.

2.5.5 Forwarding-Modi

Store-and-Forward	**Cut-Trough**
• Wird von jedem Switch beherrscht. • Speichern und weiterleiten • Checksummenprüfung (CRC) • Wirft falsche Frames raus	• Wird von besseren Switchs beherrscht. • Sehr schnell • Direktes weiterleiten • Reines Switching • Keine Fehlerüberprüfung, um Zeit zu sparen.

2.6 Switch Stack



Stack = Stapel

Man kann mehrere Switch stapeln. Diese werden dann miteinander verbunden. Bei Cisco-Switches geschieht dies auf der Vorderseite mit Glasfaser-Kabel. Damit entsteht ein grosser Switch. Die Verbindung nennt man Backplane.

2.7 VLAN (Virtual LAN)

Auf einer physischen LAN-Verkabelung werden mehrere virtuelle LANS gebildet. Jedes VLAN hat eine eindeutige Nummer. Die Pakete mit der ID 222 werden nur an die Switch Ports ausgegeben, die sich im VLAN 222 befinden.

Trunk: bündel von mehreren VLANs. Es werden alle Inputs auf diesem Port ausgegeben. Ein Kabel für alle VLANs. Oft wird dies gemacht, um mehrere VLAN-Switches miteinander zu verbinden.

2.8 Spanning-Tree-Protocol

Wir befinden uns auf dem Layer 2.

Spanning-Tree ist ein Standardverfahren, um Ringe (loops) automatisch zu unterdrücken. Dadurch kann man Backup-Leitungen schalten, die durch dieses Verfahren automatisch aktiv bzw. inaktiv geschaltet werden.

Es darf zu einer Zeit zwischen zwei Endpunkten immer nur ein Pfad existieren. Wenn ein Pfad ausfällt, bemerkt dies das STP und aktiviert darauf einen anderen funktionierenden Pfad.

Für die Kommunikation werden BPDU-Pakete verschickt. Diese sind Multicast adressiert.

2.8.1 Vorteile

- Es entstehen keine Loops im LAN
- Es sind Redundanzen möglich

- Es wird der Kostengünstigster (Kürzester) Weg zum Ziel ermittelt und genutzt
- Es ist für den Benutzer transparent

2.8.2 Vorgang

1. Die Bridges werden eingeschalten.
2. Alle Ports der Bridges werden auf blocked gestellt. Bei diesem Modi werden nur noch BPDU-Pakete entgegengenommen.
3. Jede Bridge meint, sie sei die Root-Bridge und erzählt dies den anderen Bridges.
4. Die Bridge mit der tiefsten Bridge-ID (Bridge-Priorität und MAC-Adresse) gewinnt.
5. Die Root-Bridge teilt es allen mit, dass Sie die auserwählte ist.
6. Jede Bridge bestimmt nun den Port zur Root-Bridge (Root-Port). Dieser wird mittels Pfadkosten ermittelt. Wenn die Kosten gleich hoch sind, dann wird nach der tiefsten Bridge-ID gegangen. (Kosten siehe Tabelle unten).
7. Die Designated-Bridge wird vom LAN her bestummen. Das LAN wählt eine Bridge mit den niedrigsten Pfadkosten.

2.8.3 Pfadkosten

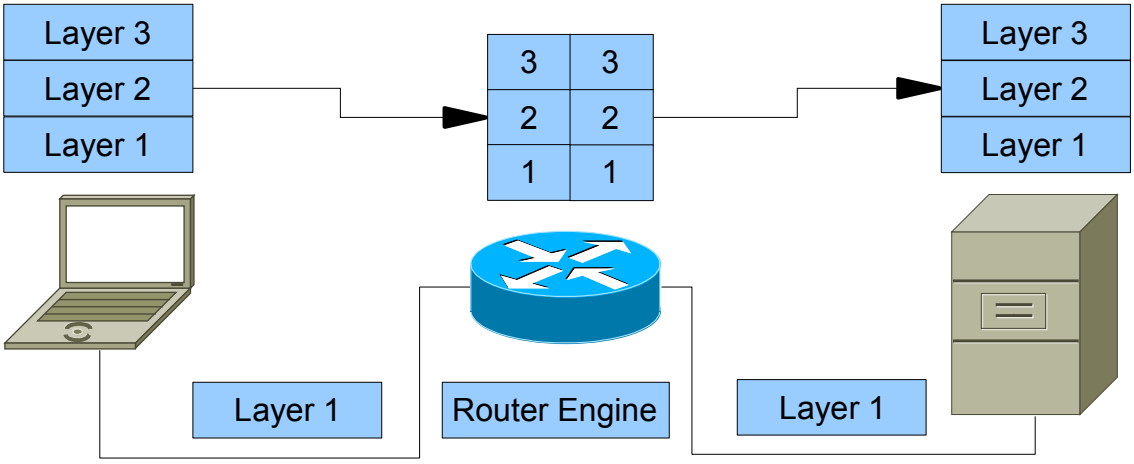
Die Pfadkosten errechnen sich wie folgt:

4 MB/S	250
10 MB/s	100
16 MB/s	62
100 MB/s	19
1 GB/s	4
10 GB/s	2

3 Layer 3: Network Layer

3.1 Adressierung auf Layer 3

Das Routing beschäftigt sich mit dem finden von Ziel- und Richtungspfad und der Weiterleitung in ein anderes Subnetz.



Der Client schickt ein IP-Paket zum Server. Der Client schaut, welche MAC-Adresse der Router hat. Der Router ist der Default-Gateway. Das IP-Paket wird mit der MAC-Adresse an den Router geschickt. Dieser leitet das Frame mit Hilfe der DST-MAC-Adresse an den Server weiter. Somit bekommt der Server das IP-Paket.

Die DST-IP# ist und bleibt immer das zu wünschende Ziel. Nur dazwischen werden die Layer 2 Frames verändert. Dort steht jeweils eine andere SRC und DST-Mac#.

3.2 Minimalkonfiguration Client

Bei einem Client ist folgendes die Minimalkonfiguration:

- IP-Adresse
- Subnet-Maske
- **Default-Gateway**

3.3 Routing-Tabelle

Subnetzadresse	Subnetmaske	Next-Hop
Wie heisst die Adresse vom Subnet	Wie heisst die Subnetmaske?	Wohin muss ich das Paket weiterleiten?

```
emanuel@discordia:~$ route
Kernel-IP-Routentabelle
Ziel            Router          Genmask         Flags Metric Ref    Use Iface
link-local      *               255.255.0.0     U     1000   0        0 wlan0
```

```
10.0.0.0        *               255.0.0.0       U     0      0        0 wlan0
default         10.0.0.1        0.0.0.0         UG    0      0        0 wlan0
```

Router kommunizieren mit Routingprotokollen untereinander und handeln einen optimalen Weg aus.

3.4 Default Gateway

Der Default-Gateway wird genutzt, wenn wir aus dem aktuellen Subnetz ein Paket senden wollen. Der Host merkt, dass das Ziel nicht im selben Subnetz liegt und sendet deshalb das Paket an den Default-Gateway. Dies ist ein Router, welcher das Paket in das richtige Subnetz weiterleitet.

3.5 IP-Paket[4]

Auf der Internetschicht des TCP/IP-Protokollstapels, auf der das IP-Protokoll arbeitet, werden die Datenpakete, wie bereits erwähnt, als Datagramme bezeichnet. Um die Datenübertragung mithilfe des IP-Protokolls genau zu erläutern, soll an dieser Stelle zunächst der IP-Header vorgestellt werden. Er enthält die Steuerdaten, die das IP-Protokoll zu einem Datenpaket hinzufügt, das ihm vom übergeordneten Transportprotokoll übergeben wird.

Der IPv4-Protokoll-Header wird wie das gesamte Protokoll in RFC 791 definiert. Seine Länge beträgt mindestens 20 Byte, dazu können bis zu 40 Byte Optionen kommen.

Byte	0	1	2	3
0	Version / IHL	Type of Service	Paket-Gesamtlänge	
4	Identifikation		Flags	Fragment-Offset
8	Time to Live	Protokoll	Header-Prüfsumme	
12	Quell-Adresse			
16	Ziel-Adresse			
20	Optionen			Padding
...	evtl. weitere Optionen			

- **Version** (4 Bit): Die Versionsnummer des IP-Protokolls, die das Paket verwendet. Bei IPv4, wie der Name schon sagt, die Version 4.
- **IHL** (4 Bit): Internet Header Length; die Länge des Internet-Headers in 32-Bit-Worten (entsprechen den Zeilen in der obigen Tabelle). Der kleinste mögliche Wert beträgt 5.
- **Type of Service** (8 Bit): Ein Code, der die Art des Datenpakets bestimmt. Bestimmte Sorten von Paketen, etwa für den Austausch von Routing- oder Status-Informationen, werden von bestimmten Netzen bevorzugt weitergeleitet. In ihrem 1999er-Aprilscherz bot die Computerzeitschrift c't ein angebliches Tool zum Download an, das diese Quality-of-Service-Informationen manipulieren könne, um die Geschwindigkeit von Internet-Verbindungen zu erhöhen. [Die Satire war immerhin so überzeugend gemacht, dass ein Leser per empörtem Leserbrief sein Abo kündigte, weil er mit derart »unmoralischem Verhalten« im Netz nichts zu tun

4 Quelle: Galileo Computing: IT-Handbuch für Fachinformatiker (ISBN 978-3-8362-1015-7)

haben wollte.]

- **Paket-Gesamtlänge** (16 Bit): Die Gesamtlänge des Datagramms in Bytes, Header und Nutzdaten.
- **Identifikation** (16 Bit): Ein durch den Absender frei definierbarer Identifikationswert, der beispielsweise das Zusammensetzen fragmentierter Datagramme ermöglicht.
- **Flags** (3 Bit): Kontrollflags, die die Paketfragmentierung regeln. Das erste Bit ist reserviert und muss immer 0 sein, das zweite (DF) bestimmt, ob das Paket fragmentiert werden darf (Wert 1) oder nicht (0), das dritte (MF) regelt, ob dieses Paket das letzte Fragment (0) ist oder ob weitere Fragmente folgen (1).
- **Fragment-Offset** (13 Bit): Dieser Wert (angegeben in 64-Bit-Blöcken) legt fest, an welcher Stelle in einem Gesamtpaket dieses Paket steht, falls es sich um ein Fragment handelt. Das erste Fragment oder ein nicht fragmentiertes Paket erhält den Wert 0.
- **Time to Live** (8 Bit): Der TTL-Mechanismus sorgt dafür, dass Datagramme nicht endlos im Internet weitergeleitet werden, falls die Empfängerstation nicht gefunden wird. Jeder Router, der ein Datagramm weiterleitet, zieht von diesem Wert 1 ab; wird der Wert 0 erreicht, leitet der betreffende Router das Paket nicht mehr weiter, sondern verwirft es.
- **Protokoll** (8 Bit): Die hier gespeicherte Nummer legt fest, für welches Transportprotokoll der Inhalt des Datagramms bestimmt ist. Die beiden wichtigsten Transportprotokolle (TCP und UDP) werden im nächsten Abschnitt beschrieben.
- **Header-Prüfsumme** (16 Bit): Die Prüfsumme stellt eine einfache Plausibilitätskontrolle für den Datagramm-Header zur Verfügung. Ein Paket, dessen Header-Prüfsumme nicht korrekt ist, wird nicht akzeptiert und muss erneut versendet werden.
- **Quelladresse und Zieladresse** (je 32 Bit): Die IP-Adressen von Absender und Empfänger. IP-Adressen wurden oben ausführlich behandelt.
- **Optionen** (variable Länge): Die meisten IP-Datagramme werden ohne zusätzliche Optionen versandt, da Absender- und Empfänger-Host sowie alle auf dem Weg befindlichen Router die jeweils verwendeten Optionen unterstützen müssen. Zu den verfügbaren Optionen gehören unter anderem Sicherheitsfeatures und spezielle Streaming-Funktionen.

4 Subnetze und Blöcke

Gegeben: Hostadresse (192.17.8.39) und Subnetzmaske (255.255.255.224)

Gesucht: Subnetzadresse, Broadcastadresse, Blockgrösse, benachbarte Subnetze

4.1 Rechnerisch

4.1.1 Subnetzadresse

- Hostadresse AND Subnetzmaske

Hostadresse	**11**000000	000**1**000**1**	0000**1**000	00**1**00111	192.17.8.39
Subnetzmaske	**11**111111	111**1**111**1**	1111**1**111	11**1**00000	255.255.255.224
Hostaddr. AND Subnetmaske	**11**000000	000**1**000**1**	0000**1**000	00**1**00000	**192.17.8.32**

4.1.2 Broadcastadresse

- Subnetzadresse OR Negierte_Subnetzmaske

Subnetzadresse	**11**000000	000**1**000**1**	0000**1**000	00**1**00000	**192.17.8.32**
Negierte Subnetzmaske	00000000	00000000	00000000	00011111	**0.0.0.31**
Subnetzadresse OR N_MSK	**11**000000	000**1**000**1**	0000**1**000	00**111111**	**192.17.8.63**

4.1.3 Blockgrösse

Wie viele Adressen in Subnetz, nicht Hosts!

- $2^{\text{Anzahl Host Bits}}$
- Negierte Subnetzmaske + 1

Subnetzadresse	11000000	00010001	00001000	00100000	**192.17.8.32**
Negierte Subnetzmaske	00000000	00000000	00000000	000**11111**	**0.0.0.31**

Anzahl Host-Bits = 5 // Blockgrösse = 2^5 = 32
Negierte Subnetzmaske + 1 = 31 + 1 = 32

4.1.4 Benachbarte Subnetze

- Benachbartes Subnetz = Subnetzadresse +/- Blockgrösse

Subnetz danach: 192.17.8.32 – 32 = **192.17.8.0**

Subnetz vorher: 192.17.8.32 + 32 = **192.17.8.64**

Dies ist jeweils die erste Adresse von diesem Subnetz. Dieses ist dann auch wieder die selbe Blockgrösse gross.

4.2 Subnetting in 60 Sekunden („Master“-Tabelle[5])

Folgende Tabelle erleichtert einem das Leben deutlich. Man ist damit sehr schnell. Der mathematische Weg wird jedoch nicht erläutert.

SN-Bits	1	2	3	4	5	6	7	8
Anz Subnetze	2	4	8	14	32	64	128	256
Subnetting	128	192	224	240	248	252	254	255
Blockgrösse	128	64	32	16	8	4	2	1
Anz. Hosts	126	62	30	14	6	2	0	-
Bitstelle	*8*	*7*	*6*	*5*	*4*	*3*	*2*	*1*

Bei alten Implementierungen ist jeweils das erste und das letzte Subnetz reserviert. Dies ist jedoch heute nicht mehr üblich.

4.2.1 Vorgehen

4.2.1.1 Subnetzadresse

- Wie lautet der Subnetzteil oder wie viele Bits sind im Subnetzteil gesetzt? (224)
- Man schaut nur noch diese Spalte an!!!
- Wir sehen die Blockgrösse (32)
- Wir schauen wie auf einer Skala, wo die IP ist. (39)
- 32 ist das grösste vielfache von 32, dass unter 39 ist.
- Das ist also der 2. Block (da der 1. Block bei 0 beginnt). Deshalb: 0 + 32 = 32. Das ist zugleich die IP vom nächsten Subnetz. (xxx.32)

4.2.1.2 Broadcastadresse

- Man nimmt die Subnetzadresse vom nächsten Block und subtrahiert 1

4.2.1.3 Blockgrösse

- Das steht schon schwarz auf weiss!

5 Inspiriert von: Millenia Magiera (aus dem Internet) // ausgearbeitet von Emanuel Duss

4.2.1.4 Benachbarte Subnetze

- Simpel: Man zählt vom aktuellen Subnetz die Blockgrösse dazu oder subtrahiert diese.

> Wenn man diese Tabelle beherrscht, darf man sich sicherlich freuen!!!

5 Routing

5.1 Routing-Tabelle bestimmen

- Alle Subnetze aufschreiben
- Die dazugehörige Subnetzmaske notieren. (Classful und Classless beachten!)
- Über welchen Router in meinem Netz ist das Subnetz „angeschlossen“. Das Interface von diesem Router auf „meiner Seite“ hinschreiben.

5.2 IP-Pakete weiterleiten

- Man nimmt die Destination IP-Adresse.
- Diese wird mit der ersten Subnetzmaske der Routing-Tabelle übereinandergelegt.
- Anhand der Subnetzmaske der Routing-Tabelle errechnen oder „erlugen aus der Tabelle“ die Subnetzadresse.
- Stimmt diese Subnetzadresse mit der Subnetzadresse der Routing-Table überein, kann diese passen.
    - Es kann jedoch sein, dass noch eine weitere Route übereinstimmt. Deshalb gehen wir ALLE EINTRÄGE der Tabelle durch!
    - Wenn mehrere übereinanderstimmen, nimmt man die „more specific“ bzw. eifach die „grössere“ Subnetzadresse (mit der kleineren Blockgrösse). (Das gibt weniger Broadcast :D – oh ja – schön.)
- Wenn quasi keine übereinanderstimmt, ausser 0.0.0.0, dann wird diese genommen. Diese stimmt immer und wird auch Default-Route genannt.
- Nun schauen wir beim passenden Eintrag, an welchen Next-Hop das IP-Paket weitergeleitet werden soll. Dies machen wir auch so.

> Kurz gesagt: Netzwerkadresse erstellen und vergleichen. Wenns passt, nimmt man den Next-Hop beim Eintrag mit der „grössten“ Subnetzmaske.

5.3 Supernetting / Summieren / Summarizing

Damit kann man mehrere Routing-Tabellen-Einträge zu einem Eintrag zusammenfassen.

- Man erstellt eine Tabelle: [Netzwerkadresse] [Subnetzmaske] [Start- und End-IP]
- Entsteht ein zusammenhängender Bereich **ohne Lücken**? Falls JA, schreibt man diesen Bereich hin.

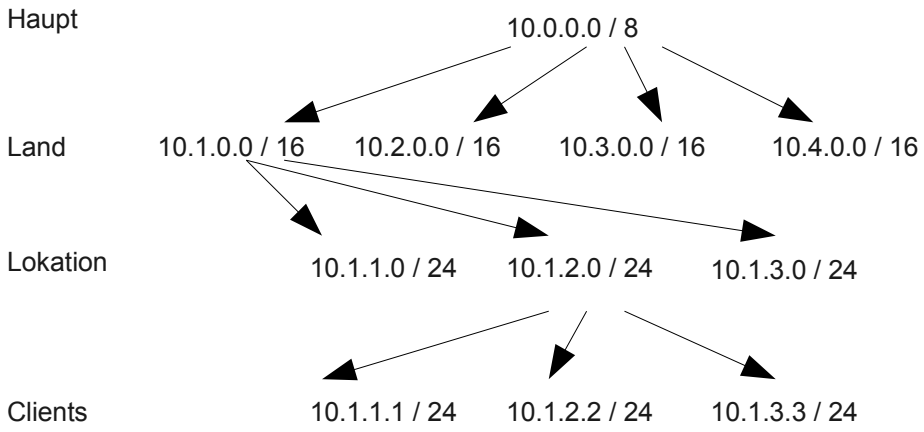
160 – 164	164 – 168	168 – 172	172 – 176
160 – 168			
			170-176
160 – 176 (Netz, das alle anderen beinhaltet!)			

5.4 Hierarchisches IP-Konzept erstellen

Ein Hierarchisches IP-Konzept ist eine Methode / Schema IP-Adressen zu vergeben. Dabei schaut man auf Lokationen und teilt diese so praktisch ein. Eine IP-Adresse könnte folgendermassen aufgebaut sein:

- [Land].[Kanton].[Lokation].[Fortlaufende Nummer]

Wenn man ein IP-Konzept erstellt, dann Summiert man quasi rückwärts.



Der Router beim Haupt hat nicht für jedes Subnetzwerk auf den Lokationen einen Routing-Eintrag. Er hat legidlich für jedes Land einen Eintrag. Auf dem Hauptrouter wurde also Subnettiert, sprich es wurden die Subnetze (Lokationen) auf den Länern summiert.

5.4.1 Vorteile

- Mehr Übersicht
- Man sieht sofort, woher eine IP-Adresse kommt
- Das Summieren geht einfacher

5.5 Routingtabellen intepretieren

Kurz und knapp zum Zeichnen des Netzes anhand der Routing-Tabelle:

- Man zeichnet zuerst den Host auf mit allen Interfaces (Schnittstellen).
- Dann macht man jeweils das Netzwerkziel an das entsprechende Interface dran.

5.6 NAT (Network Adress Translation)

So sieht eine NAT-Tabelle aus:

Indide-IP-Adresse	Outside-IP-Adresse	Destination Port
192.168.1.1	83.78.118.43	23
192.168.1.23	83.78.118.43	22
192.168.1.5	83.78.118.44	80
192.168.1.3	83.78.118.59	5900

Bei Home-Routern wäre in der Mitte immer die selbe Globale WAN-IP eingetragen. Es können nur mehrere gebraucht werden, wenn man mehrere „gekauft“ (?) hat bzw. ein kleines Subnetz zur Verfügung hat.

--- WTF ---

5.6.1.1 Private IP-Adressen / Reservierte IP-Adressen

0.0.0.0	All Networks
255.255.255.255	All Hosts
127.0.0.1 / 8	Localhost
127.0.0.0 / 8	Localnet
169.254.10.0 – 169.254.XXXXXXXXXXXXXXXX	APIPA (Autokonfiguration bei Problemen)

6 Supernetting / Summieren / Summarizing

7 Das IP-Paket

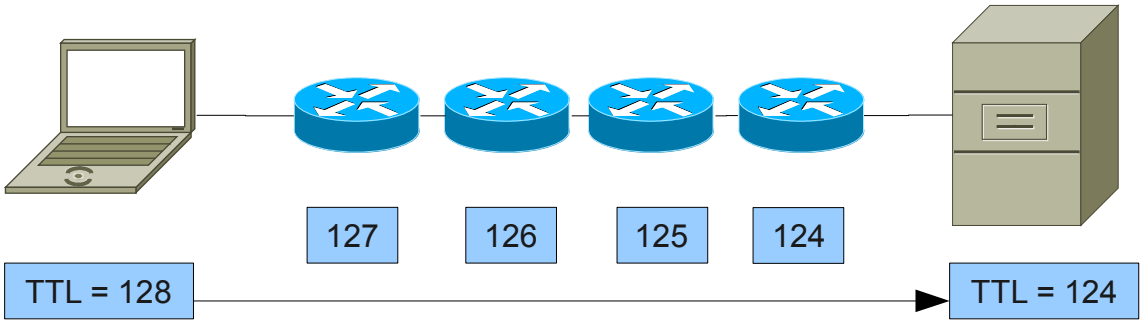
7.1.1 IP-Adressen

- 32-Bit = 4 x 8 Bit

7.1.2 Schreibweise als dezimale oktetteÜberblick

0	*4*	*8*	*12*	*161*	*24*	*28*	*31*
Version	IHL	TOS		Total Lengh			
Identification				Flags	Fragment Offset		
Time To Live		Protocol		Header Checksum			
Source Adress							
Destination Adress							
Options and Padding							

7.1.3 TTL



Das TTL beschreibt die Gültigkeitsdauer eines IP-Pakets. Das TTL-Feld ist standardmässig auf 128 eingestellt. Das TTL-Feld besagt, wie oft das Paket bei einem Hop (Router) vorbeikommen darf. Jeder Router verringert das Paket um 1. Wenn dann ein Router ein Paket erhält, von dem das TTL-Feld 0 ist, wirft er es weg.

Stichwortverzeichnis