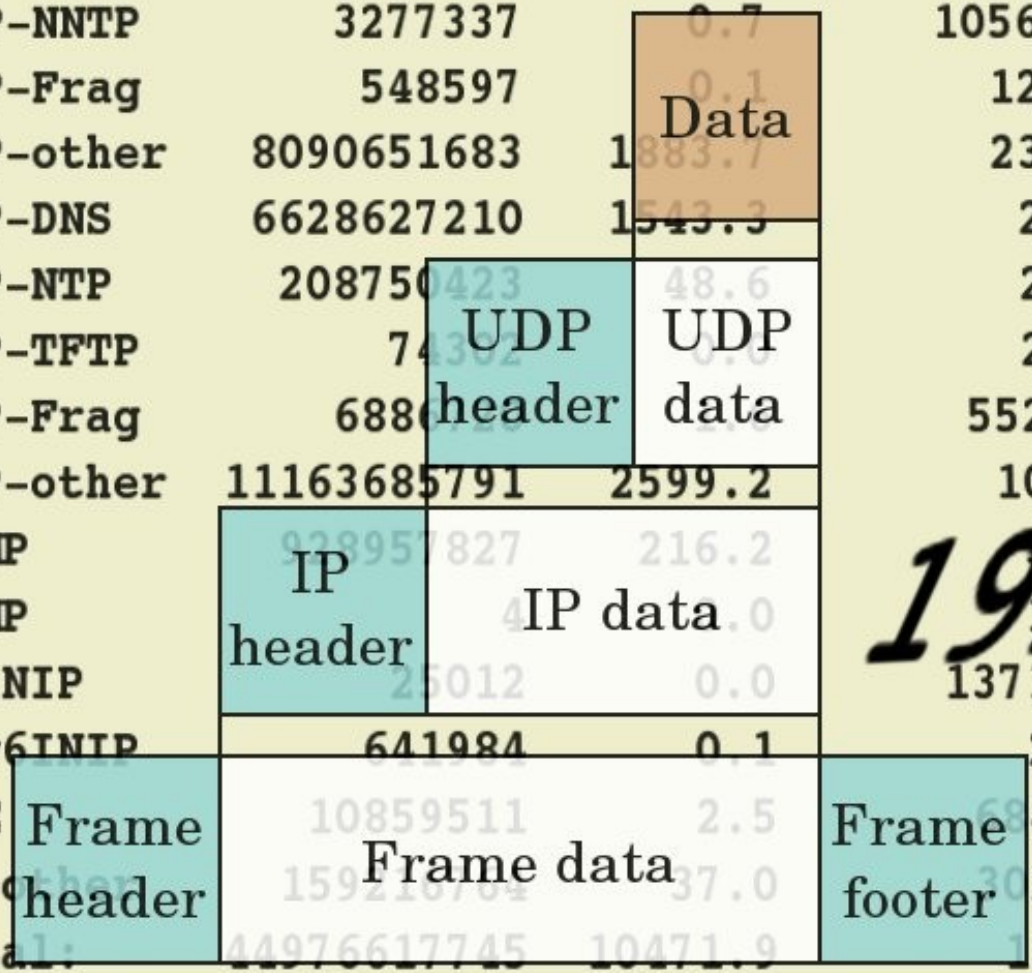
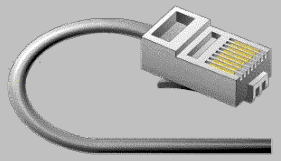
Protocol	Total Flows	Flows /Sec	Packets /Flow	Bytes /Pkt	Packets /Sec	Active(Sec) /Flow	Idle(Sec) /Flow
TCP-Telnet	3484677	0.8	87	111	70.9	23.6	22.7
TCP-FTP	127336721	29.6	20	66	599.2	7.0	27.6
TCP-FTPD	46489384	10.8	138	875	1493.7	2.1	29.3
TCP-WWW	15115628719	3519.3	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
TCP-SMTP	2433340539	566.5	12	414	6939.4	6.1	26.1
TCP-X	43955166	10.2	[illegible]	[illegible]	[illegible]	[illegible]	31.5
TCP-BGP	4179374	0.9	36	86	35.2	96.0	16.8
TCP-NNTP	3277337	0.7	1056	886	805.9	43.8	22.9
TCP-Frag	548597	0.1	12	170	1.6	7.1	25.6
TCP-other	8090651683	1883.7	23	547	44836.9	8.0	24.9
UDP-DNS	6628627210	1543.3	2	77	3290.9	2.0	30.0
UDP-NTP	208750423	48.6	2	74	101.0	3.0	29.8
UDP-TFTP	74302	0.0	2	214	0.0	1.3	29.3
UDP-Frag	688[illegible]	[illegible]	552	588	886.1	34.5	23.9
UDP-other	11163685791	2599.2	10	272	28167.1	4.2	29.1
ICMP	938957827	216.2	[illegible]	77	788.2	5.4	28.0
IGMP	[illegible]	[illegible]	1	40	0.0	0.0	22.2
IPINIP	25012	0.0	1371	137	7.9	43.8	21.2
IPv6INIP	641984	0.1	2	387	0.3	4.4	37.9
GRE	10859511	2.5	[illegible]4	425	1731.7	73.3	19.3
IP-[illegible]	159210764	37.0	[illegible]2	426	11222.5	56.3	21.0
Total:	44976617745	10471.9	[illegible]6	538	167638.2	4.6	25.9

TCP/IP Protokoll Suite
Skript Kapitel 5ff



192.0.2.5/24



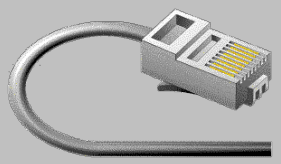
TCP/IP Protocol Suite

TCP/IP Protokoll-Sammlung

Überblick

welche Protokolle gehören dazu
welche Aufgaben haben diese Protokolle
welche Schichten decken diese Protokolle ab

Kapitel 4 / Seite 32

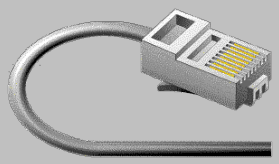


TCP/IP Protocol Suite

Die TCP/IP Protokoll Suite ist wie das OSI-Modell Layer basiert.

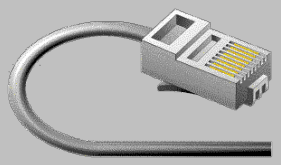
Die Layer 1 und 2 sowie die Layer 5 bis 7 sind jeweils zusammen gefasst

OSI Modell		TCP Modell
7	Application	Application Layer
6	Presentation	
5	Session	
4	Transport	Transport Layer
3	Network	Network Layer
2	Data-Link	Network Interface Layer
1	Physical	



TCP/IP Protocol Suite

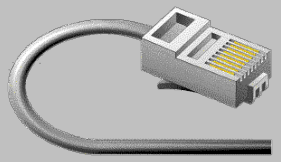
OSI Modell		TCP Modell	Protokolle
7	Application	Application Layer	HTTP, FTP, SMTP, POP3, IMAP4, Telnet, ssh, DNS, NTP, BGP,
6	Presentation		
5	Session		
4	Transport	Transport Layer	TCP, UDP, ICMP, ...
3	Network	Network Layer	IP
2	Data-Link	Network Interface Layer	ARP, RARP
1	Physical		



TCP/IP Protocol Suite; Layer 2

ARP: ARP wird verwendet um die Ethernet MAC-Adresse einer lokalen IP-Adresse zu finden.

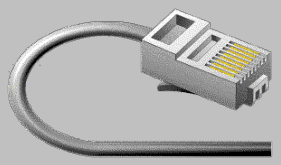
RARP: Ist eine Methode um einer Station eine IP anhand ihrer MAC-Adresse zu zuweisen. (RARP ist ein Vorgänger von BOOTP/DHCP)



TCP/IP Protocol Suite; Layer 3

IP: Internet Protocol (**IP**) ist ein **verbindungs-loses** Protokoll, das verwendet wird, um Paket vom Transport Layer durch das Netz zu leiten.

IP ist ein routebares Protokoll.

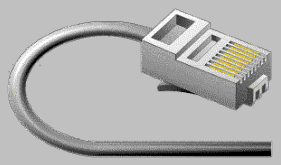


TCP/IP Protocol Suite; Layer 4

Im IP-Header ist eine Protokoll-Nummer enthalten, die angibt welches Protokoll die Daten im IP Paket angehören (analog dem Ethertype-Feld im Ethernet Header).

Protokoll-Nummern sind in der Datei `/etc/protocols` [1] abgelegt. Die Nummern werden von der IANA [2] vergeben und sind für alle IP Protokolle (IPv4 und IPv6) gültig.

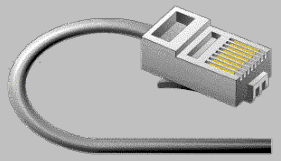
```
[1] windows: .../system/drivers/etc/protocol
[2] http://www.iana.org/assignments/protocol-numbers
```



TCP/IP Protocol Suite; Layer 4

ICMP: Internet Control Message Protocol (**ICMP**) (Protocol #1) ist ein **verbindungsloses** Protokoll, das verwendet wird um Information-, Status- oder Fehler-Meldungen zu übermitteln.

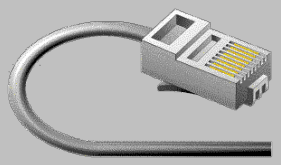
Mögliche Meldungen: Echo Reply, Destination unreachable, Source Quench, Echo Request, Time Exeeded, ...



TCP/IP Protocol Suite; Layer 4

TCP: Transmission Control Protocol (**TCP**) (Protocol #6) ist ein **verbindungsorientierts** Protokoll, das verwendet wird, um Daten gesichert durchs Netz zu transportieren.

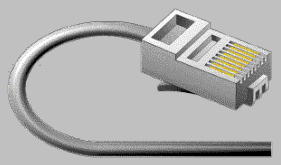
TCP garantiert, dass die Daten – in der gleichen Reihenfolge wie gesendet – ankommen.



TCP/IP Protocol Suite; Layer 4

UDP: User Datagram Protocol (**UDP**) (Protocol #17) ist ein **verbindungsloses** Protokoll, das verwendet wird, um Daten mit geringem Overhead durch das Netz zu transportieren.

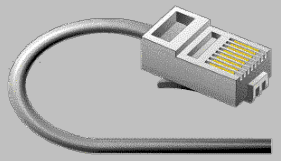
UDP kennt weder Flusskontrolle noch Fehlerkorrektur. Die Anwendungen müssen das selber erledigen!



TCP/IP Protocol Suite; Layer 4

Andere: Es gibt noch viele andere IP-Protokolle auf Layer 4.

Diese werden für spezielle Anwendungen verwendet:
Routing Protokolle (OSPF, EIGRP, ...),
IP-SEC (Encap Security Payload, Authentication Header),
vrrp, ...
Es ist für Firewall Konfigurationen nützlich zu wissen dass auch andere IP Protokolle als ICMP, UDP und TCP existieren!

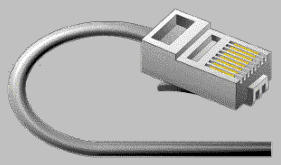


TCP/IP Protocol Suite; Layer 7

Applikationen:

Unzählige Applikationen verwenden IP ...

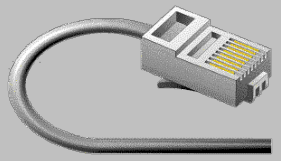
Jede Applikation muss – leider – selber dafür besorgt sein, dass der Empfänger die Daten richtig interpretieren kann!



TCP/IP Protocol Suite; Layer 7

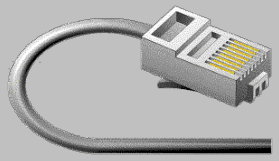
Applikationen:

Datentransfer:	FTP, TFTP, NFS, SMB/CIFS,
DruckDienste:	IPP, LPD
RemoteDienste:	Telnet, ssh, X11, RDP, ...
eM@il / News:	SMTP, POP3, IMAP4, NNTP
Web:	HTTP
Netzwerkdienste:	DHCP, BOOTP, DNS, NTP, Syslog, ...
Tunnel:	IPSEC, IPinIP, IPv6inIP, ...



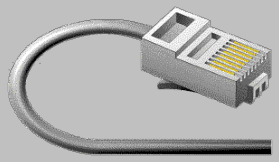
Verteilung der Protokolle

Protocol	Total Flows	Flows /Sec	Packets /Flow	Bytes /Pkt	Packets /Sec	Active(Sec) /Flow	Idle(Sec) /Flow
TCP-Telnet	2142021	0.4	22	163	11.1	34.4	25.4
TCP-FTP	69292174	16.1	25	57	417.7	5.7	26.9
TCP-FTPD	27762305	6.4	75	833	485.7	2.6	34.6
TCP-WWW	7684202068	1789.1	18	664	33002.4	4.1	29.9
TCP-SMTP	945875329	220.2	13	448	2985.4	7.2	29.1
TCP-BGP	2462947	0.5	24	101	13.8	112.5	19.4
TCP-NNTP	2043881	0.4	989	904	470.8	42.1	28.6
TCP-Frag	376749	0.0	4	250	0.3	5.5	30.6
TCP-other	4417559690	1028.5	18	525	19002.4	8.9	29.9
UDP-DNS	2088078323	486.1	2	76	1228.0	3.2	33.8
UDP-NTP	95496552	22.2	1	76	35.1	2.6	34.3
UDP-TFTP	165665	0.0	4	96	0.1	18.3	29.0
UDP-Frag	3223364	0.7	797	651	598.4	35.9	30.3
UDP-other	4579162977	1066.1	11	329	12189.0	4.9	32.9
ICMP	315821611	73.5	3	75	281.4	7.9	32.0
IGMP	9	0.0	1	34	0.0	5.9	32.0
IPINIP	5416	0.0	65	397	0.0	78.9	32.4
IPv6INIP	1482	0.0	4	313	0.0	1.4	42.3
GRE	3006667	0.7	609	393	426.7	106.7	23.2
IP-other	52845052	12.3	273	469	3364.7	67.6	25.1
Total:	**20308186365**	**4728.3**	**15**	**542**	**74554.4**	**5.6**	**31.0**

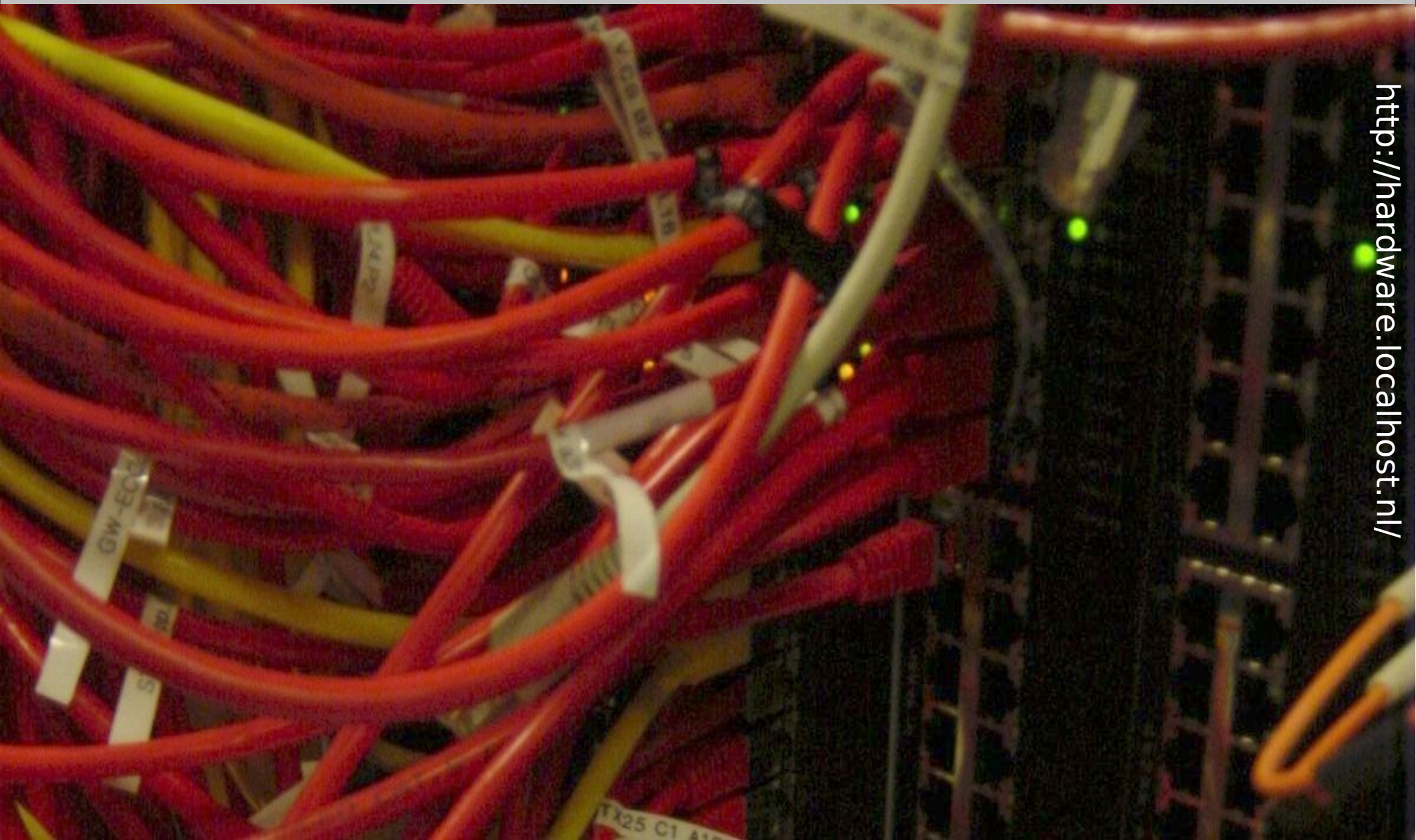


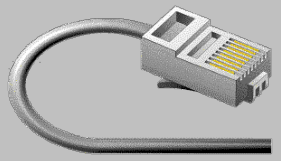
Paket Grösse

```
IP packet size distribution (720011M total packets):
   1-32    64    96   128   160   192   224   256   288   352
  .003 .395 .074 .038 .022 .011 .041 .006 .009 .004

   320   384   416   448   480   512   544   576  1024  1536
  .004  .004  .004  .004  .006  .006  .005  .015  .034  .306

  2048  2560
  .000  .000
```



Fragen ?

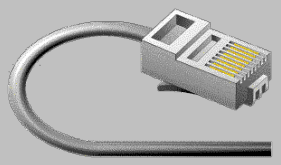




Internet Protokoll IP

Ziele

- Was sind routebare Protokolle
- IP-Adressen
  - Netzwerk-, IP-Adressen, Netzmasken
  - IP Mathematik



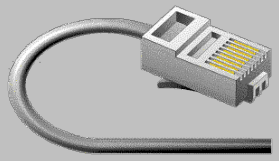
Routebare Protokolle

≠ Routing Protkolle!

Routebare Protokolle können zwischen lokalen und entfernten Adressen unterscheiden

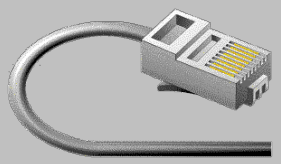
Rouetbare Protokolle kennen eine Netz-Hierarchie
Protokolle: IPv4, IPV6, Appletalk, IPX, ...

Nicht routebaren Protokollen fehlen diese Eigenschaften!
Protokolle: NetBEUI



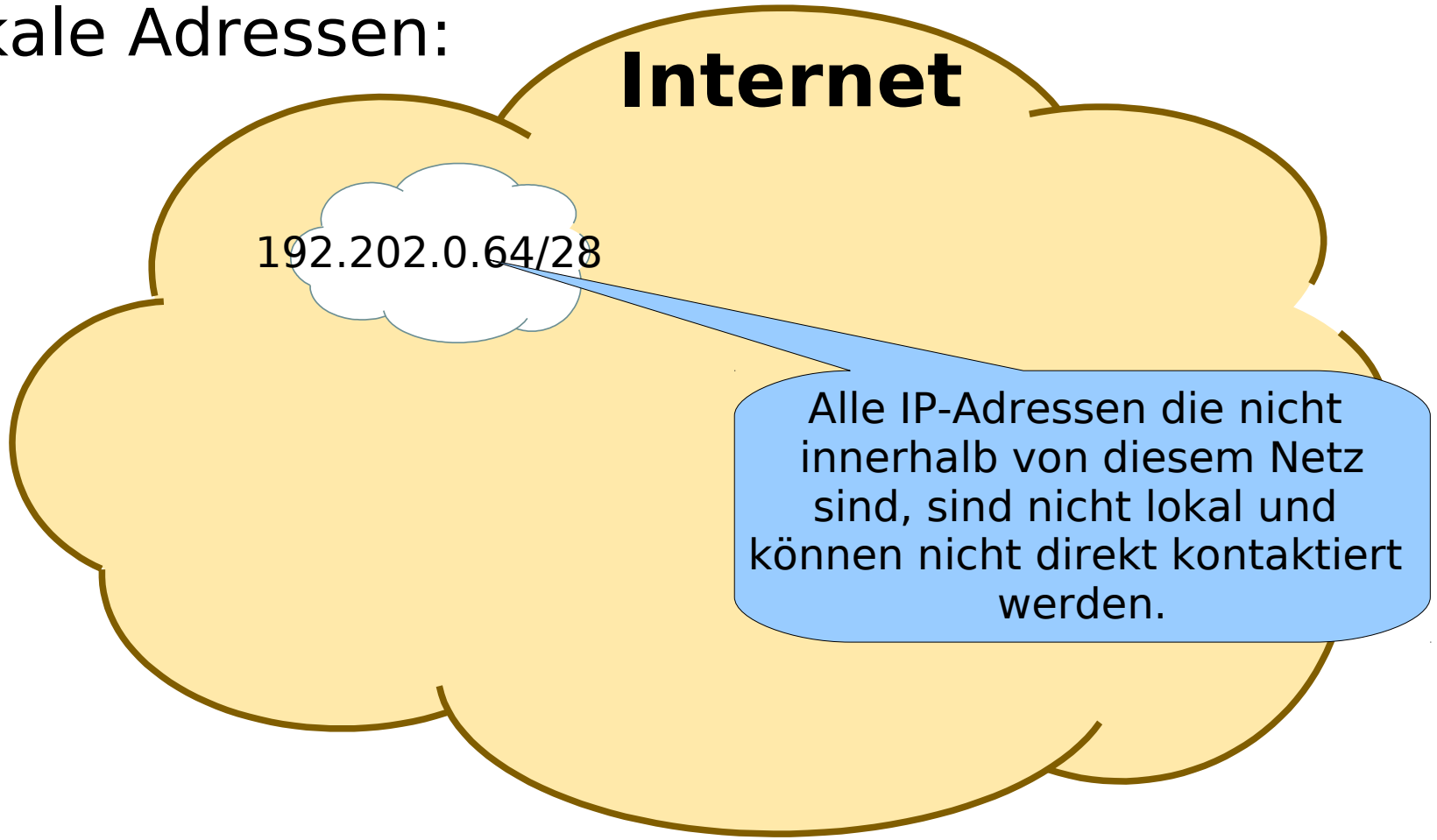
Internet Protokoll

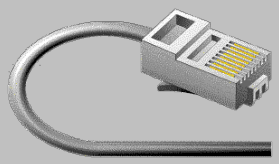
- Das Internet Protokoll IPv4 bzw. IPv6 ist routbar.
- Routebare Protokolle unterteilen die Adressen in einen Netz-Teil und einen Host-Teil
- Der Host-Teil ist **lokal**, **direkt** erreichbar
- Der Netz-Teil ist **nur** via einem speziellen Gerät (Router, Gateway) erreichbar.
- Bei IPv4 / IPv6 ist die Trennung zwischen Netz- und Host-Teil variabel
- Bei IPv4 / IPv6 zeigt die Netzmaske wo die Trennung zwischen Netz- und Host-Teil liegt



Internet Protokoll

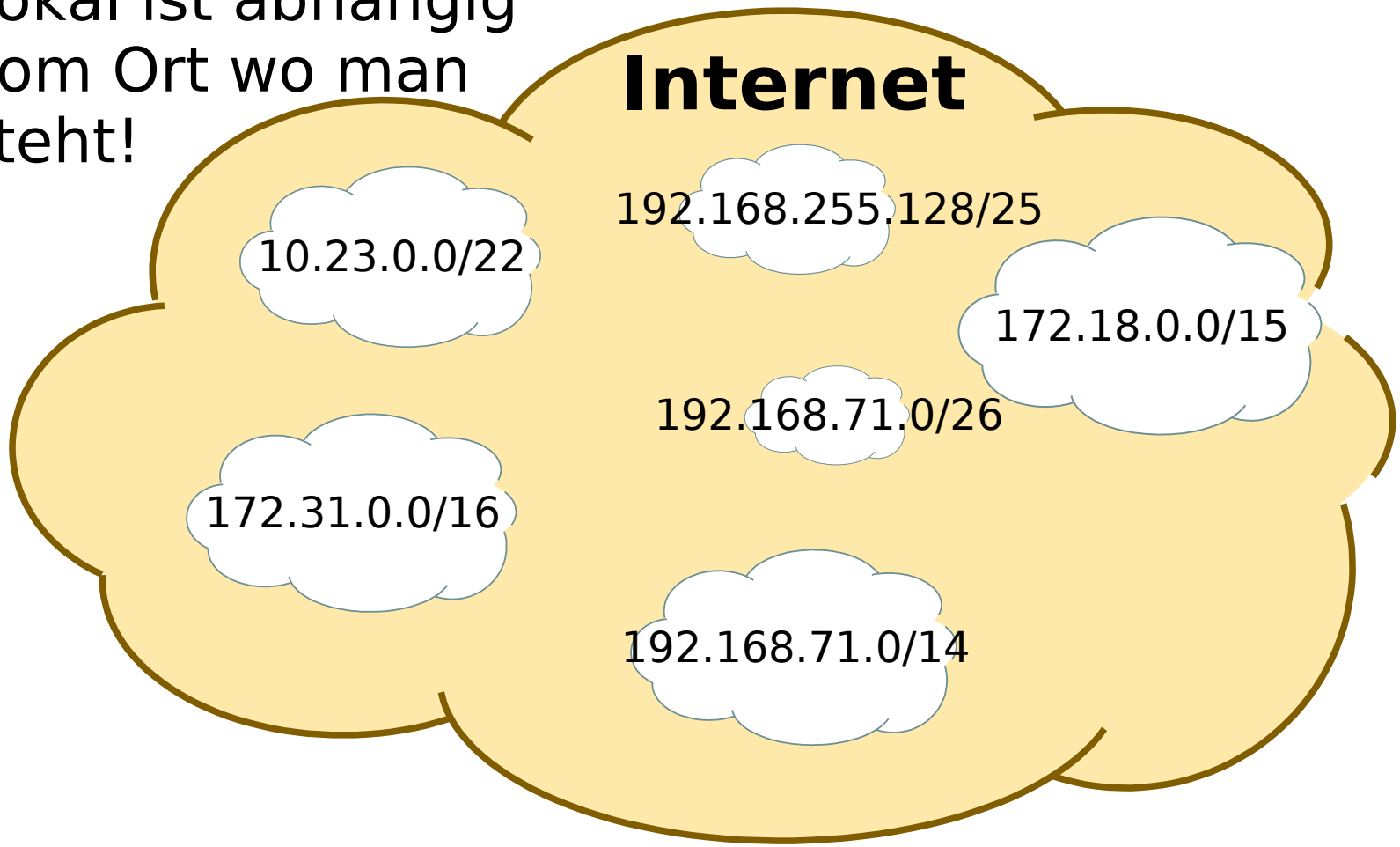
Lokale Adressen:

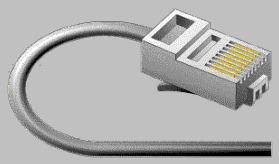




Internet Protokoll

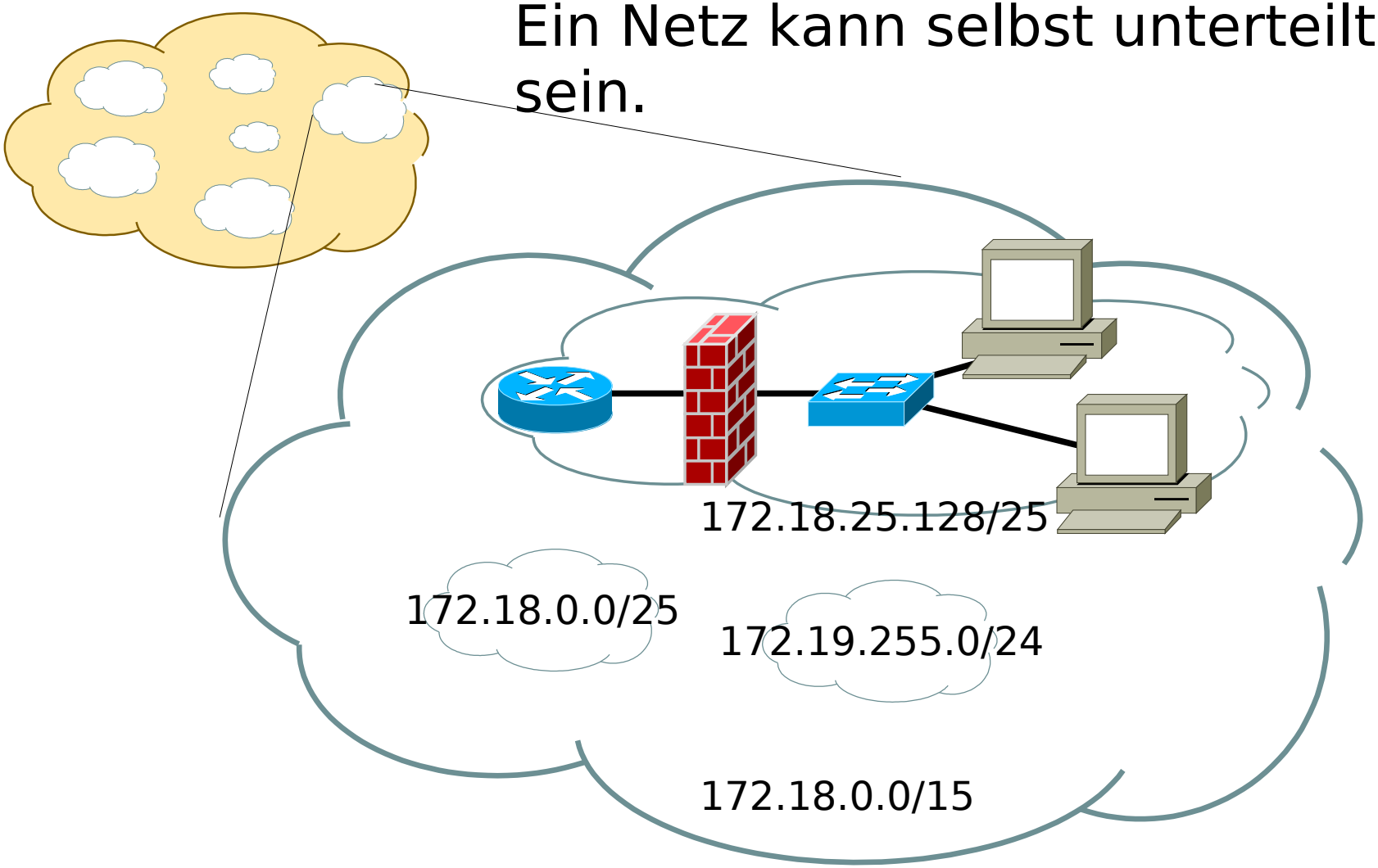
Lokal ist abhängig vom Ort wo man steht!

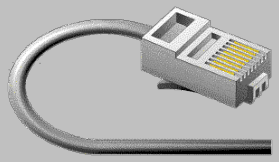




Internet Protokoll

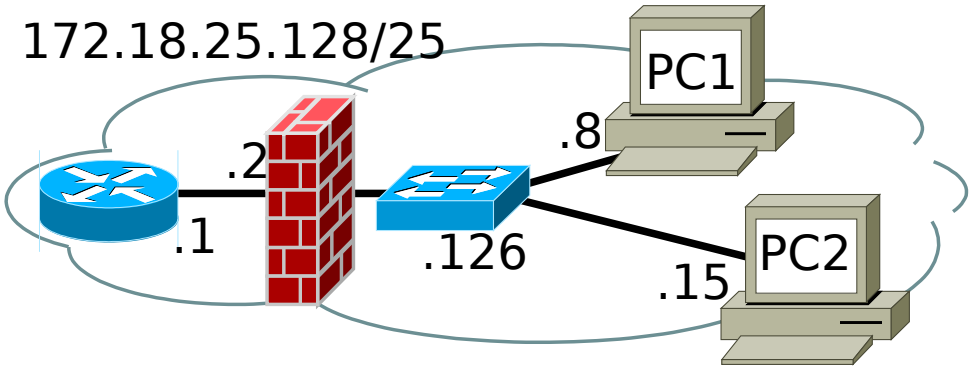
Ein Netz kann selbst unterteilt sein.





Internet Protokoll

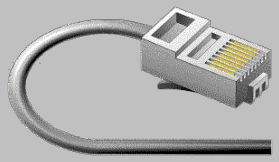
Die Host-Adresse adressiert den Host innerhalb des gegeben Netzwerkes



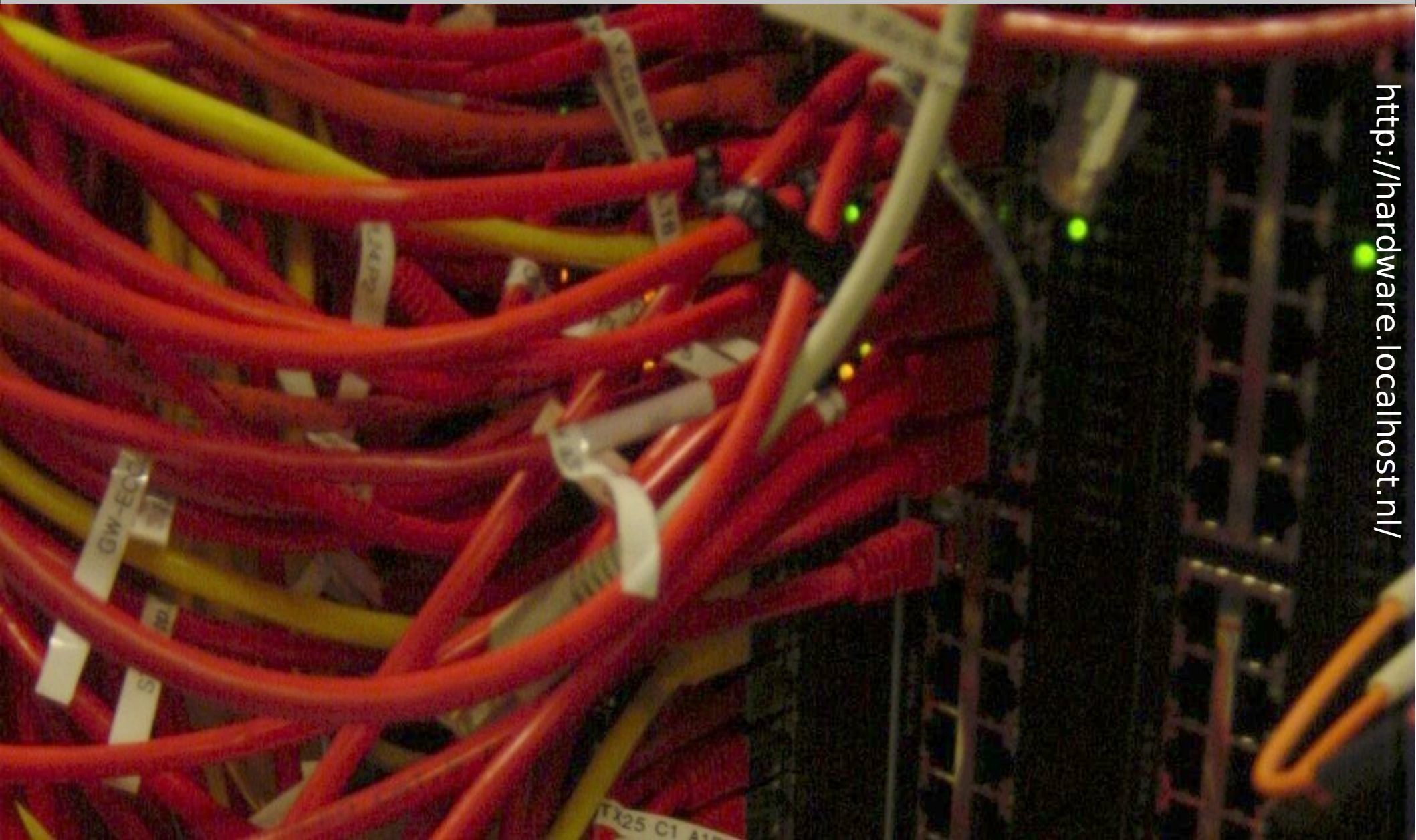
Router:	172.18.25.129/25
Firewall:	172.18.25.130/25
Switch:	172.18.25.254/25
PC1:	172.18.25.135/25
PC2:	172.18.25.143/25

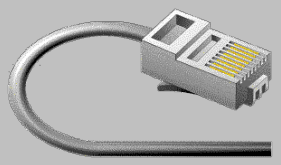
Annahme: die Firewall ist transparent.

Die IP-Adresse wird aus der Netzwerk-Adresse und der Host-Adresse zusammen gesetzt!



Fragen ?



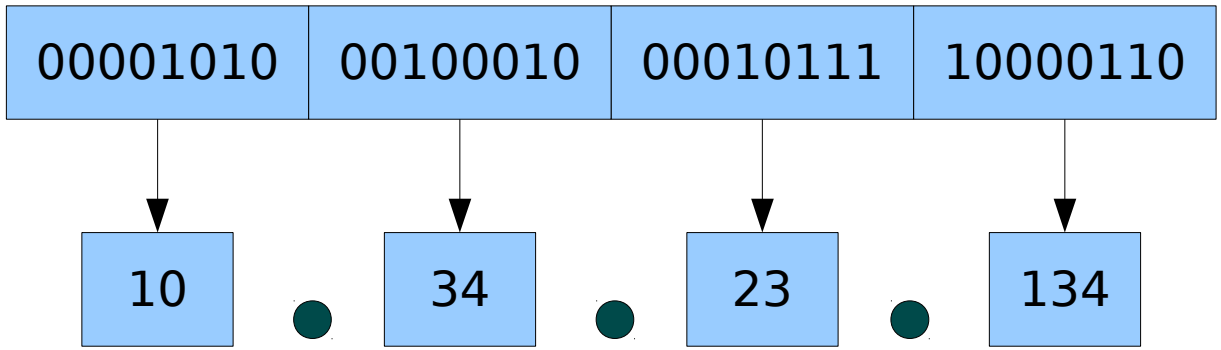


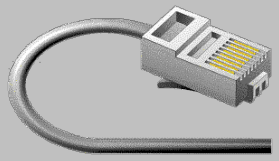
IP-Adressen

IPv4-Adressen sind 32Bit lang.

Jeweils 8Bit werden zusammen gefasst und als Dezimalzahl geschrieben.

Zwischen den Dezimalzahlen wird ein Punkt eingefügt.





Internet Protokoll

Zu Beginn wurden die Adressen in Klassen (A,B,C,D,E) unterteilt.

Die **Klassen A,B,C** werden für normale Anwendungen verwendet (mit Ausnahmen!).

Für jede Klasse (ABC) ist eine fixe Netzmaske definiert

1.0.0.0 – 126.255.255.255

128.0.0.0 – 191.255.255.255

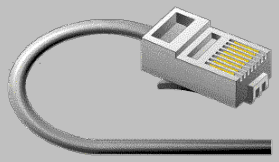
192.0.0.0 - 223.255.255.255

Die **Klasse D** ist für Multicast reserviert

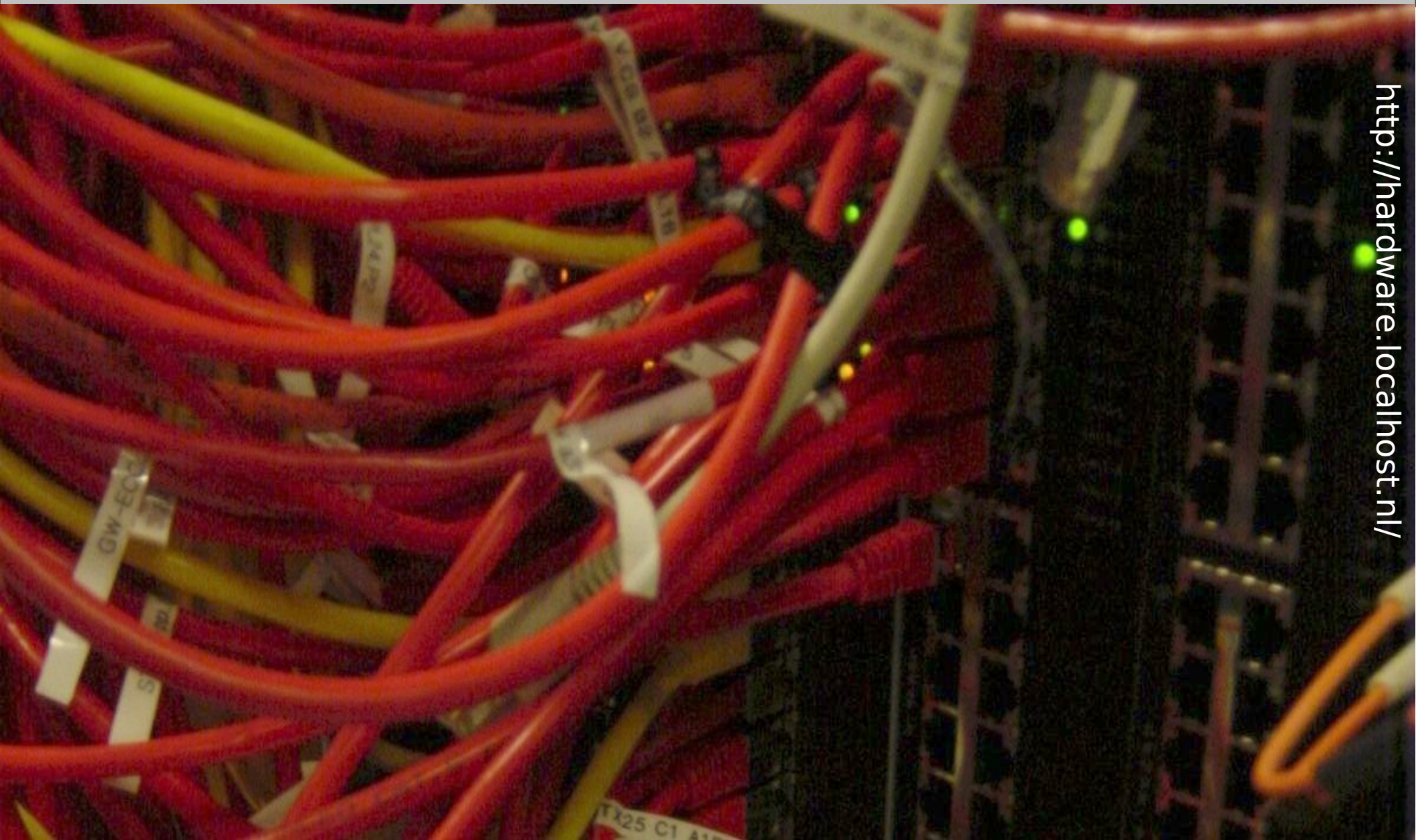
224.0.0.0 - 239.255.255.255

Die **Klasse E** ist für Experimente reserviert.

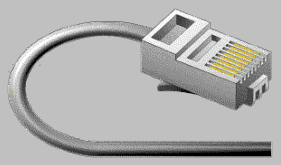
240.0.0.0 - 255.255.255.255



Fragen ?

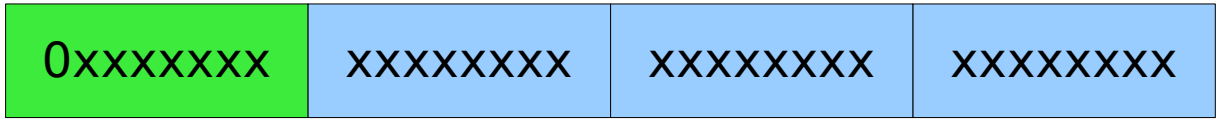


http://hardware.localhost.nl/



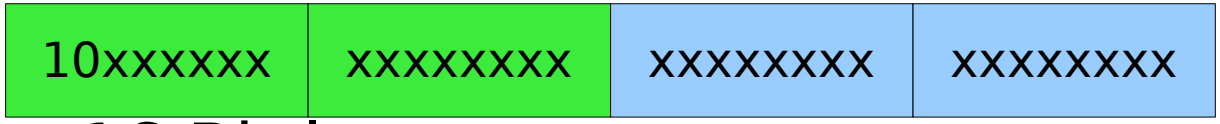
IPv4 Adressen – Classfull

Klasse A

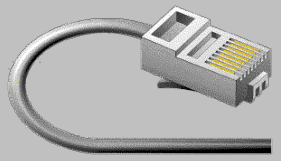


Die Netzadresse ist 8 Bit lang.
Die Hostadresse ist 24 Bit lang
Das erste Bit der IP-Adresse ist 0 (binär)
1.0.0.0 - 126.255.255.255
0.0.0.0 - 0.255.255.255 sowie 127.0.0.0 - 127.255.255.255 sind reserviert

Klasse B

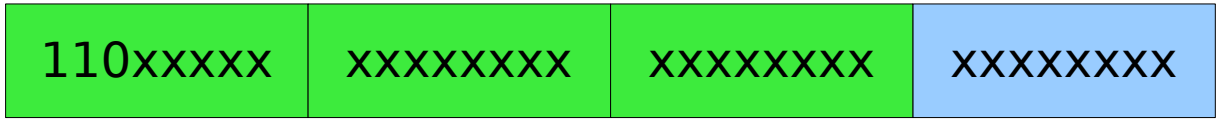


Die Netzadresse ist 16 Bit lang.
Die Hostadresse ist 16 Bit lang
Die Netz-Adresse beginnt mit 10 (binär)
128.0.0.0 - 191.255.255.255

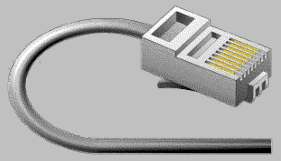


IPv4 Adressen – Classfull

Klasse C



Die Netzadresse ist 24 Bit lang.
Die Hostadresse ist 8 Bit lang
Das ersten Bits lauten 110 (binär)
192.0.0.0 - 223.255.255.255



Internet Protokoll – Classfull

Klasse D

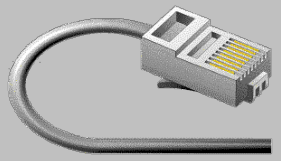
1110xxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx

Das ersten Bits lauten 1110 (binär)
verwendet für Multicasts
Es gibt keine Netzmaske!
224.0.0.0 - 239.255.255.255

Klasse E

1111xxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx

Die ersten Bits lauten 1111 (binär)
verwendet für Forschung / Experimente
Es gibt keine Netzmaske!
240.0.0.0 - 255.255.255.254

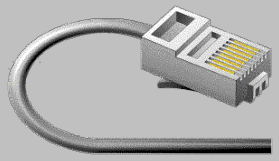


Internet Protokoll – Classfull

Übersicht

Class	Erstes Octet
A	1 - 126
B	128 - 191
C	192 - 223
D	224 - 239
E	240 - 255

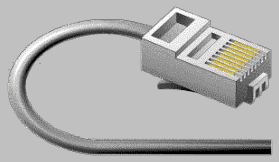
Das Netz 0.0.0.0/8 ist nicht verwendet!
Das Netz 127.0.0.0/8 ist für Loopback Interfaces und Loopback reserviert!



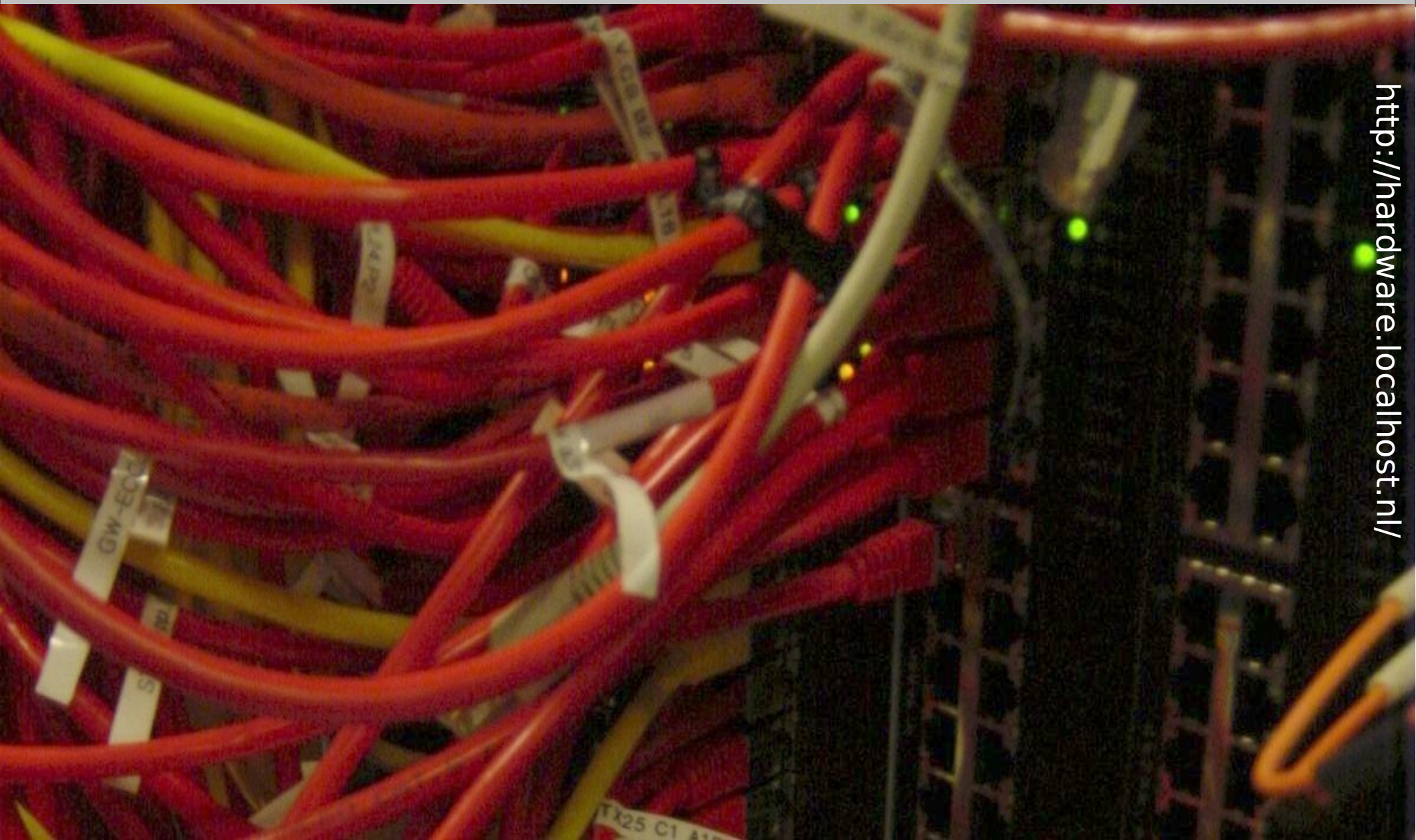
Internet Protokoll – Classfull

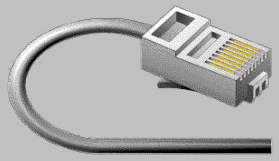
Übersicht

Class	Netze	Hosts
A	126	16'777'214
B	16'384	65'534
C	2'097-152	254
D	n/a	n/a
E	n/a	n/a



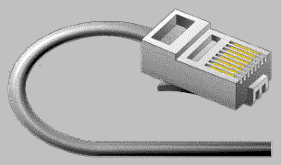
Fragen ?





Internet Protokoll

Später (ab 1993) ist man von der fixen Klassen-Einteilung abgewichen und hat das **Classless Internet-Domain Routing** (CIDR) eingeführt.

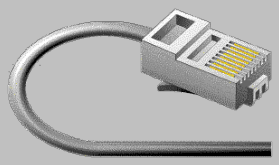


Internet Protokoll – Classless

Die starre Einteilung der Klassen hat sich als unpraktisch erwiesen. Zu viele IP-Adressen sind blockiert und können nicht verwendet werden.

Die flexiblere Lösung **Classless Internet-Domain Routing** (CIDR) wurde 1993 eingeführt. Zu jedem Netz muss die Netzmaske bzw. die Grösse des Netzes angegeben werden.

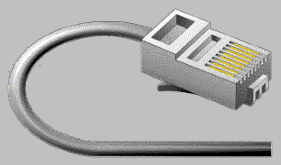
CIDR und NAT/PAT hat den Verbrauch von IPv4 Adressen verlangsamt. Trotzdem steigt der IPv4 Bedarf weiterhin an.



Internet Protokoll – Classless

CIDR erfordert, dass immer die Grösse des Netzes bzw. die Netzmaske angegeben wird – Dies ist notwendig, weil nicht mehr aufgrund der IP-Adresse alleine entschieden werden kann, wo die Grenze zwischen Netzwerk- und Host-Adresse liegt.

▶ Fehlt die Netzmaske bei einer Adresse, so wird die Netzmaske der entsprechenden Klasse angenommen werden.

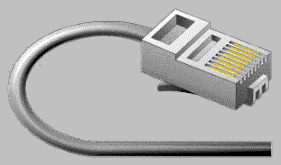


CIDR Routen im Internet

In der globalen Routingtabelle sind CIDR Netze anzutreffen:

```
Network          Next Hop            Metric LocPrf Weight Path
* i3.0.0.0        139.4.71.37          9000    110      0 702 703 80 i
* i4.0.0.0        139.4.71.37          9000    110      0 702 701 3356 i
* i4.0.0.0/9      139.4.71.37          9000    110      0 702 701 3356 i
* i4.21.41.0/24   217.6.49.129        10000    100      0 3320 2914 16467 36806 i
* i4.36.200.0/21  217.6.49.129        10000    100      0 3320 3549 14135 i
* i4.67.64.0/22   217.6.49.129        10000    100      0 3320 6453 11608 19281 i
...
* i203.81.64.0/19  139.4.71.37         9000    110      0 702 701 2914 9988 i
* i203.81.96.0/21  139.4.71.37         9000    110      0 702 701 3491 9237 i
* i203.81.104.0/22 139.4.71.37         9000    110      0 702 701 3491 9237 i
* i203.81.108.0/22 139.4.71.37         9000    110      0 702 701 3491 9237 i
* i203.81.112.0/20 139.4.71.37         9000    110      0 702 701 4725 24289 i
* i203.81.128.0/19 139.4.71.37         9000    110      0 702 703 17608 i
* i203.81.160.0/20 217.6.49.129        10000    100      0 3320 2914 9988 18399 i
...
```

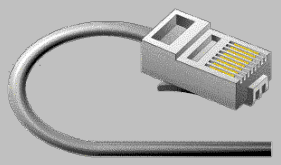
- Die ersten beiden Einträge (rot) sind classfull Routen



Internet Protokoll

Wie findet man die Netzwerk-Adresse von einer IP-Adresse?

Die Netzwerk-Adresse ist immer am Anfang der IP-Adresse.
Die Host-Adresse ist immer an Ende der IP-Adresse.
Die Netzmaske gibt die Trennstelle zwischen Netz- und Host-Adresse an.



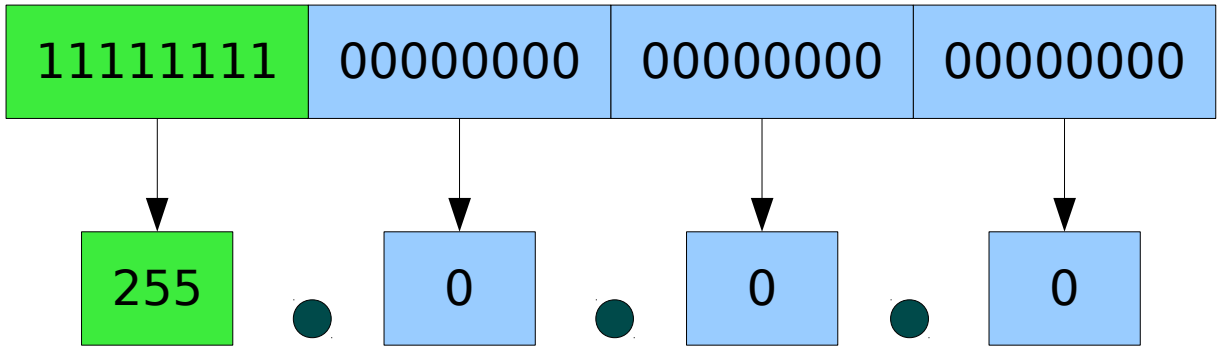
Netzmaske

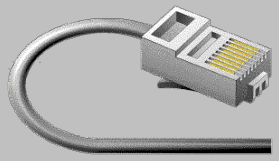
Die Netzmaske definiert wo die Grenze zwischen Netzwerk- und Host-Adresse liegt

Die Netzmaske ist wie die IPv4 Adresse 32Bit lang.

Der Netzwerk-Teil wird mit 1 markiert

Der Host-Teil wird mit 0 markiert



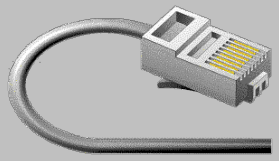


Netzmaske

Es gibt nur einen Netz- und einen Host-Teil.
Es darf nur einen Übergang von 1➔0 geben

11111111	00000000	00000000	00000000	OK
11111111	01111000	00000000	00000000	Falsch

Netzmasken können daher nur folgende Werte enthalten:
255, 254, 252, 248, 240, 224, 192, 128, 0



Netzmaske

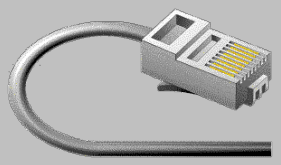
Die Netzmaske kann alternativ auch in der Slash-Notation angegeben werden.
Dazu werden die Anzahl der 1 in der Netzmaske hinter einem Slash (/) angegeben

11111111	1111	0000	00000000	00000000
11111111	00000000		00000000	00000000
11111111	11111111		11111111	11111111

12x 1 ➔ **/12**

8x 1 ➔ **/8**

32x 1 ➔ **/32**



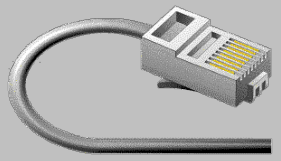
IP-Mathematik

Ist eine IP-Adresse und die dazugehörende Netzmaske bekannt, so können verschiedene Adressen berechnet werden.

$$NetzwerkAdresse = IPAdresse \wedge Netzmaske$$

$$BroadcastAdresse = IPAdresse \vee \neg Netzmaske$$

$$Anzahl\,Hosts = 2^{(32 - SlashNetzmaske)} - 2$$

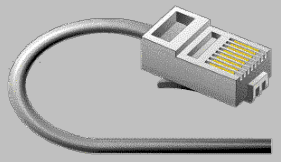


IP-Mathematik

Ein Host ist innerhalb eines Netzes, wenn seine

IP-Adresse zwischen der **Netzwerk-Adresse** und der **Broadcast-Adresse** liegt:

Netzwerk-Adresse < IP-Adresse < Broadcast-Adresse

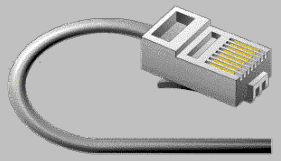


IPv4 Konfigurations Regeln

Die Netzwerk-Adresse zusammen mit der Netzmaske definiert ein IP-Netz eindeutig.

Die **Netzwerk**- oder die **Broadcast**-Adresse darf bei keinem Host konfiguriert werden!

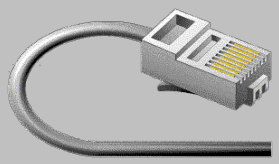
Pakete, die an die Broadcast Adresse gesendet werden, werden von allen Rechnern bearbeitet. Darum darf – wie die Netzwerk-Adresse – die Broadcast-Adresse keinem Rechner zugewiesen werden!



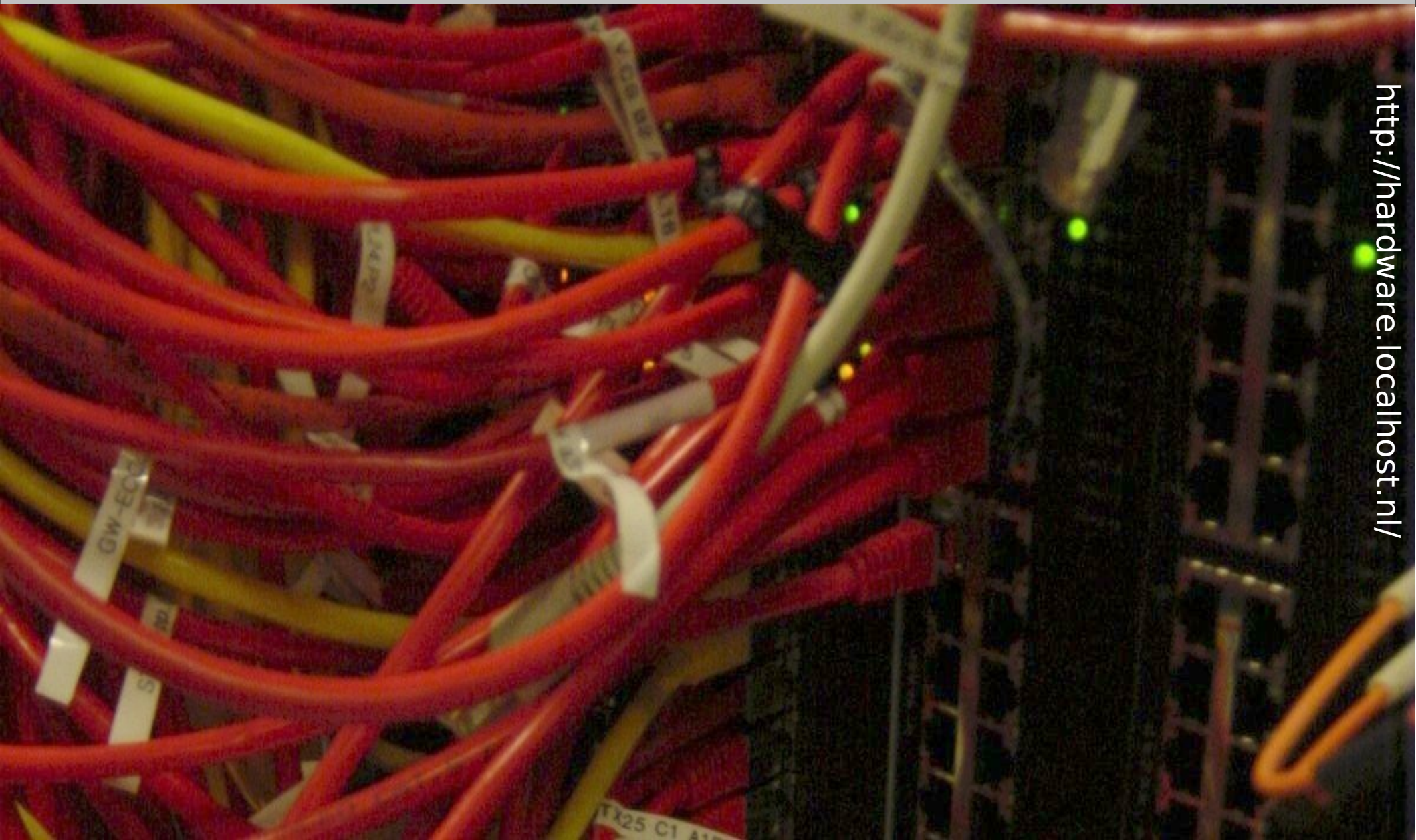
IPv4 Konfigurations - Regeln

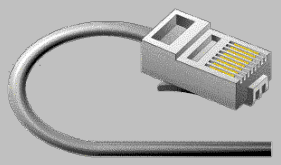
Jeder Host muss innerhalb eines Netzwerkes eine eindeutige IP-Adresse besitzen.
Muss der Host global ansprechbar sein, so muss eine weltweit eindeutige IP-Adresse verwendet werden.

Doppelt vergebene IP-Adressen bereiten grosse Probleme in einem Netzwerk!
Es ist möglich ein ganze Netzwerke so lahm zu legen!!!



Fragen?





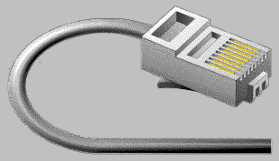
Dez ➔ Binär

34 ➔ binär?

Durch fortlaufendes Teilen kann die Zahl umgerechnet werden.

Dividend:	34	17	8	4	2	1
Divisor:		2	2	2	2	2
Resultat:	17	8	4	2	1	0
Rest:		0	1	0	0	0	1

34_{10} ➔ **$1\,0\,0\,0\,1\,0_2$**



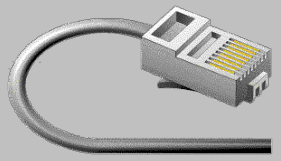
Binär ➔ Dez

$1\ 0\ 0\ 0\ 1\ 0_2$ ➔ Dezimal?

$1*2^{\wedge 5} + 0*2^{\wedge 4} + 0*2^{\wedge 3} + 0*2^{\wedge 2} + 1*2^{\wedge 1} + 0*2^{\wedge 0}$

$32+0+0+0+2+0 = 34$

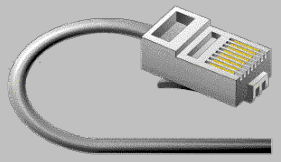
$1\ 0\ 0\ 0\ 1\ 0_2$ ➔ **34_{10}**



IP Mathematik Beispiele

Bestimmen sie die Netzwerk-Adresse von folgender IP-Adresse: 217.14.67.72 255.255.255.240

```
217.14.67.72        11011001.00001110.01000011.01001000
255.255.255.240     11111111.11111111.11111111.11110000
UND verknüpft       11011001.00001110.01000011.01000000

Zurück gewandelt: Netzwerkaddresse: 217.14.67.64/28
```

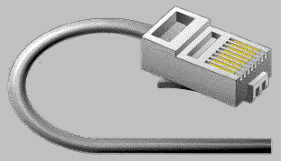


IP Mathematik Beispiele

Bestimmen sie die Broadcast-Adresse von der IP-Adresse:
217.14.67.72 255.255.255.240

```
255.255.255.240     11111111.11111111.11111111.11110000
Negiert
255.255.255.240     00000000.00000000.00000000.00001111
217.14.67.72        11011001.00001110.01000011.01001000
ODER verknüpft
Broadcast-Adresse 11011001.00001110.01000011.01001111
```

Zurück gewandelt: **Broadcast-Adresse: 217.14.67.79**



IP Mathematik Beispiele

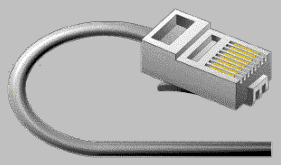
Wieviele IP-Adressen haben in folgendem Netz platz?
217.14.67.72 255.255.255.240

255.255.255.240 11111111.11111111.11111111.11110000

➔ 28 Einsen

Anzahl der Hosts = $2^{(32-28)} - 2 = 2^4 - 2 = 16 - 2 = 14$

Das Netz reicht für 14 Hosts.



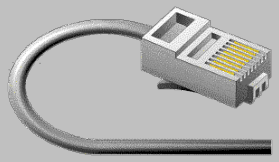
IP Mathematik Beispiele

Welche Subnetzmaske benötige ich, wenn ich in einem Netz maximal 48 Rechner unterbringen möchte:

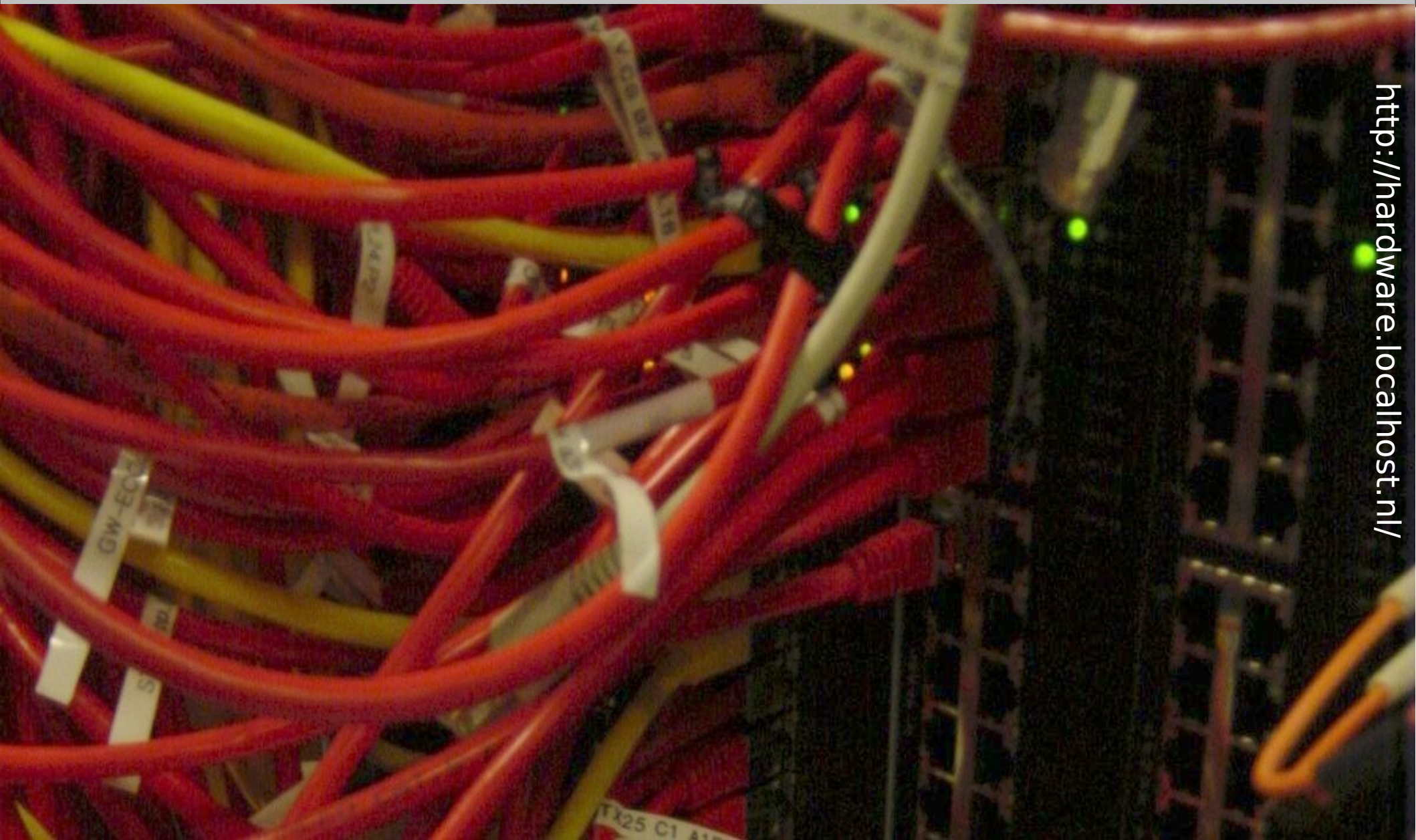
48 Rechner:
➔ 32er Netz (/27) ist zu klein (30Hosts).
➔ 64er Netz (/26) ist zu gross (62Hosts).
➔ 128er Netz (/25) ist viel zu gross (126 Hosts).

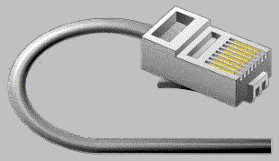
➔ für 48 Rechner ist mindestens ein 64er (/26) Netz notwendig. Die Netzmaske ist 255.255.255.192.

Achtung: Meistens braucht es auch noch eine IP-Adresse für einen Router.



Fragen?



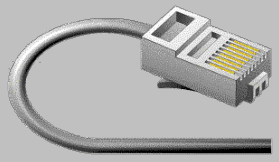


reservierte IP Adressen

Neben den Class D und Class E Adressen gibt es noch weitere IPv4-Adressen die für bestimmte Zwecke reserviert sind:

RFC 1918: Private IP-Adressen
10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16 können für private Netze verwendet werden und werden im Internet nicht geroutet.
Diese Adressen sind weltweit **nicht** eindeutig.

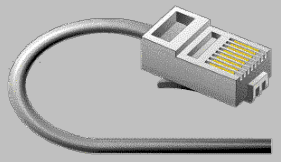
Bei VPN Tunnel zwischen gleichen privaten Netzen kann es zu komplizierten Setups kommen.



reservierte IP Adressen

0.0.0.0/8 - Addresses in this block refer to source hosts on "this" network. Address 0.0.0.0/32 may be used as a source address for this host on this network; other addresses within 0.0.0.0/8 may be used to refer to specified hosts on this network.

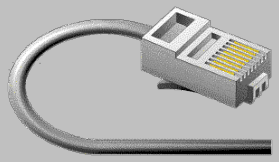
127.0.0.0/8 - This block is assigned for use as the Internet host loopback address. A datagram sent by a higher level protocol to an address anywhere within this block should loop back inside the host. This is ordinarily implemented using only 127.0.0.1/32 for loopback, but no addresses within this block should ever appear on any network anywhere.



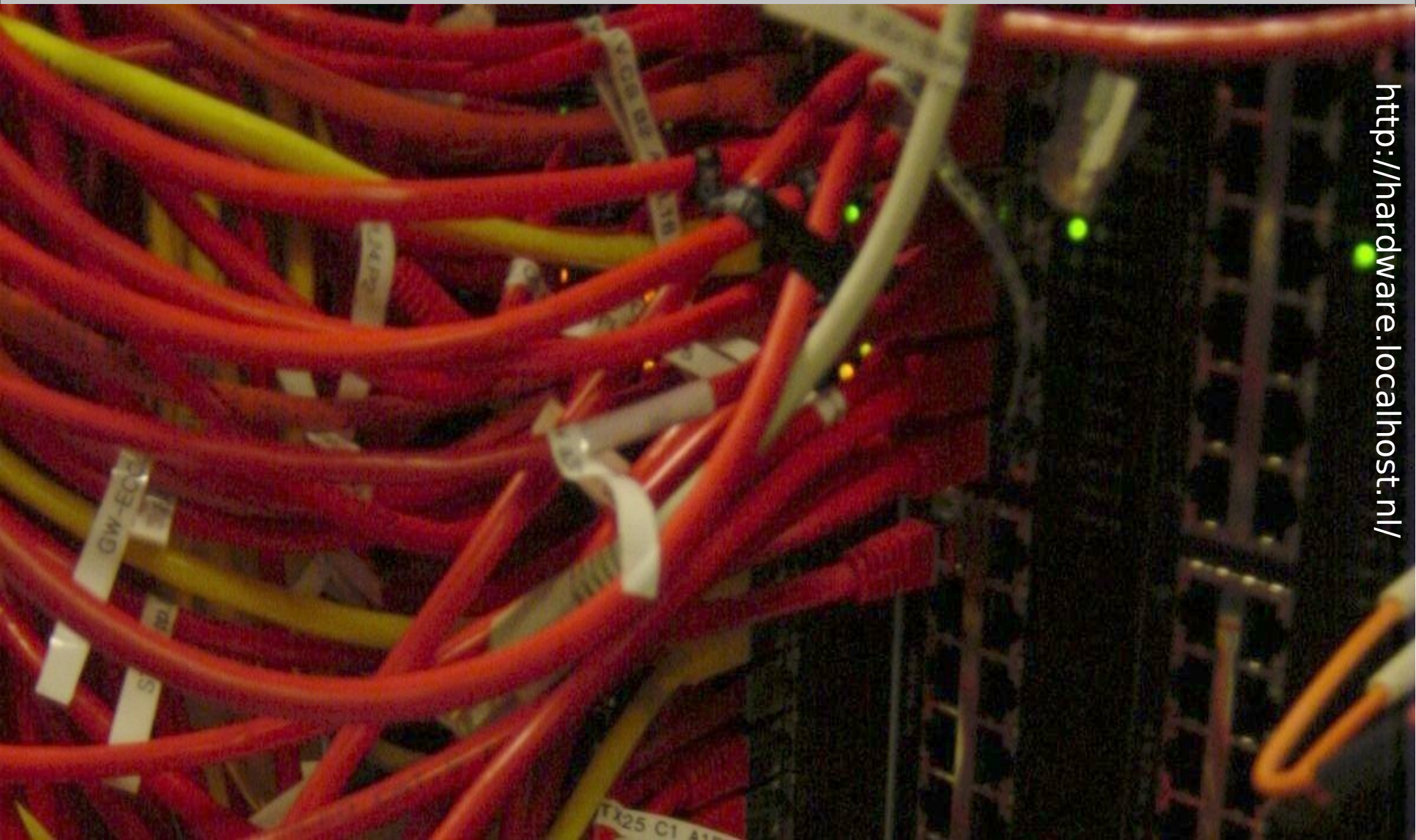
reservierte IP Adressen

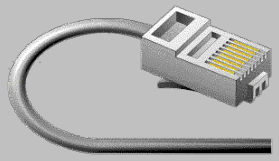
169.254.0.0/16 - This is the "link local" block. It is allocated for communication between hosts on a single link. Hosts obtain these addresses by auto-configuration, such as when a DHCP server may not be found.

192.0.2.0/24 - This block is assigned as "TEST-NET" for use in documentation and example code. It is often used in conjunction with domain names *example.com* or *example.net* in vendor and protocol documentation.
Addresses within this block should not appear on the public Internet.



Fragen?





Wer vergibt IP-Adressen

Damit das Internet funktioniert müssen die IP-Adressen weltweit koordiniert werden.

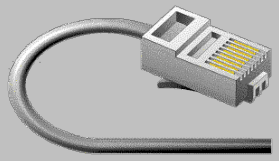
Die Vergabe der IP-Adressen erfolgt hierarchisch.

Als oberste Instanz koordiniert die Internet Assigned Numbers Authority (IANA) die IP-Adressen.

IANA vergab /8 Blocks an Regionale Internet Registraturen (RIR)[1].

IANA vergibt weiterhin IPv6 Blöcke![2]

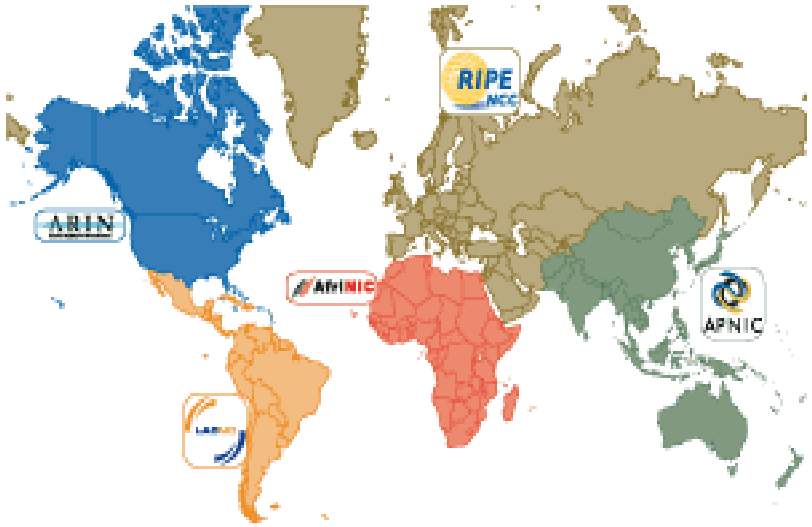
[1] http://www.iana.org/assignments/ipv4-address-space/ipv4-address-space.txt
[2] http://www.iana.org/assignments/ipv6-address-space/ipv6-address-space.txt

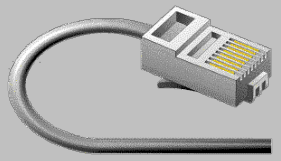


Wer vergibt IP-Adressen

Weltweit gibt es 5 RIRs:

AFRINIC	African Internet Numbers Registry
APNIC	Asia Pacific Network Information Centre
ARIN	American Registry for Internet Numbers
LACNIC	Latin American and Caribbean Internet Addresses Registry
RIPE	Réeseaux IP Européans





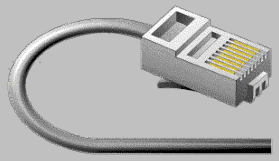
Wer vergibt IP-Adressen

Die RIRs allozieren IP-Adressen an die Local Internet Registry (LIR) – ihre Mitglieder.

Jedes LIR vergibt dann die IP-Adressen an ihre Kunden gemäss deren Bedarf unter Berücksichtigung der Richtlinien vom entsprechenden RIR.

Wer ein LIR ist, kann bei RIPE öffentlich eingesehen werden[1]

[1] http://www.ripe.net/membership/indices/

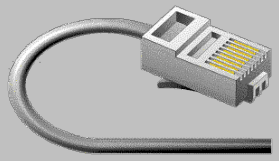


PI oder PA?

RIR vergeben 2 Arten von IP-Adressen:

PI Provider Independent
Diese Adressen sind nicht an einen bestimmten Provider gebunden. Diese Adressen müssen angefordert und gegenüber der RIR begründet werden!

PA Provider aggregatable
Diese Adressen sind an einen bestimmten Provider gebunden und können bei einem Wechsel vom Provider **nicht** mitgenommen werden!

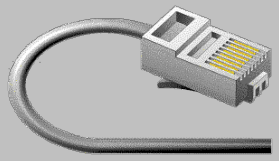


Whois

Mit dem Befehl **`whois`**[1] kann nachgesehen werden wem welche IP-Adresse zugeordnet wurde:

```
inetnum:        212.55.196.64 - 212.55.196.71
netname:        CH-MAX-MUSTER
descr:          BBI for Max Muster
country:        CH
admin-c:        MM3421-RIPE
tech-c:         MM3421-RIPE
status:         ASSIGNED PA
notify:         ripe@cyberlink.ch
mnt-by:         CYBERLINK-MNT
changed:        dvg@cyberlink.ch 20021107
source:         RIPE
```

[1] oder http://www.ripe.net/whois



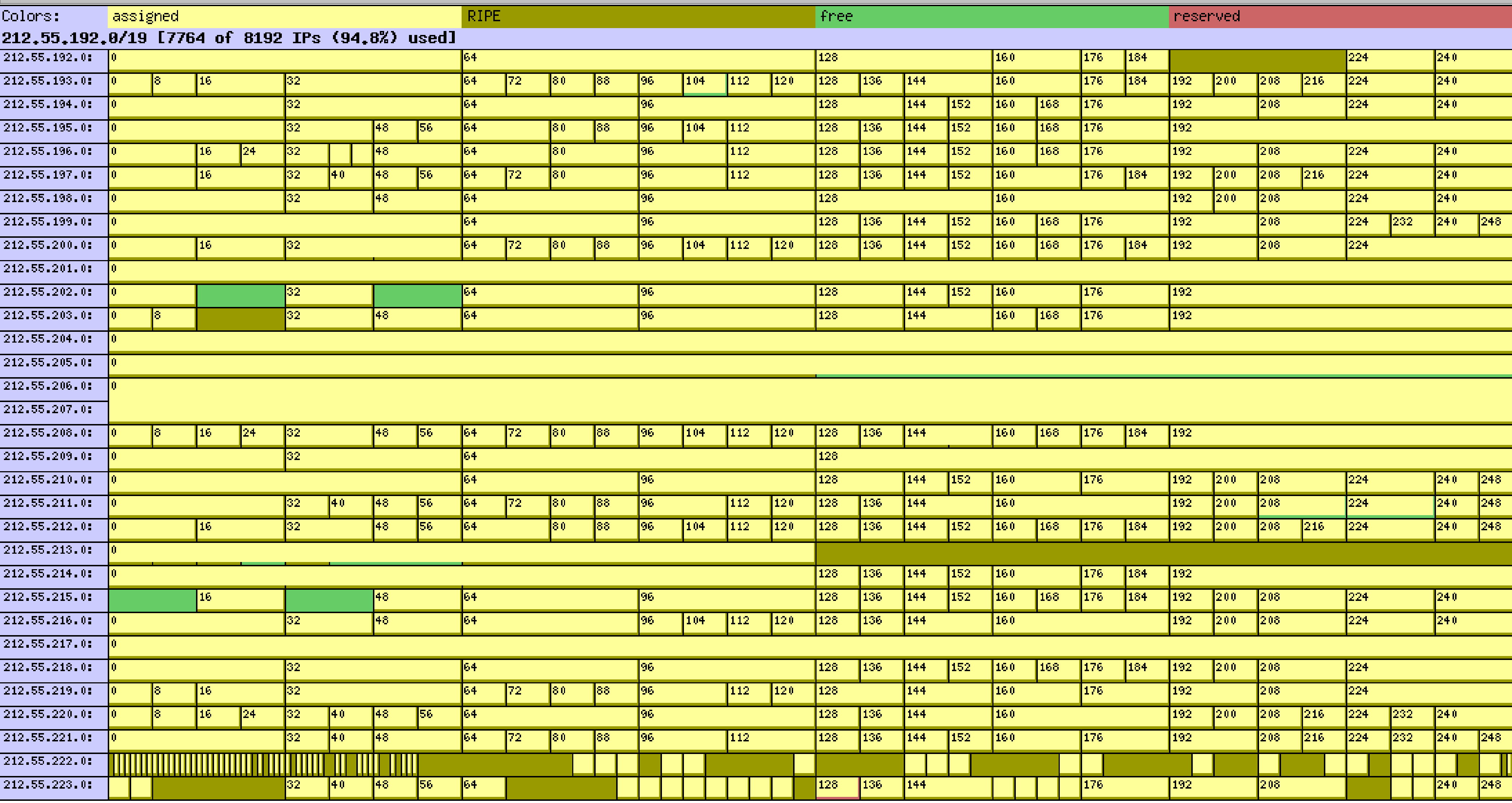
IP Adressen

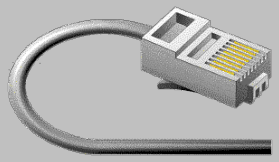
Wenn sie IP-Adressen benötigen, stellen sie sich folgende Fragen:

Müssen die/alle Rechner weltweit erreichbar sein?
nein → RFC1918 Private Adressen.

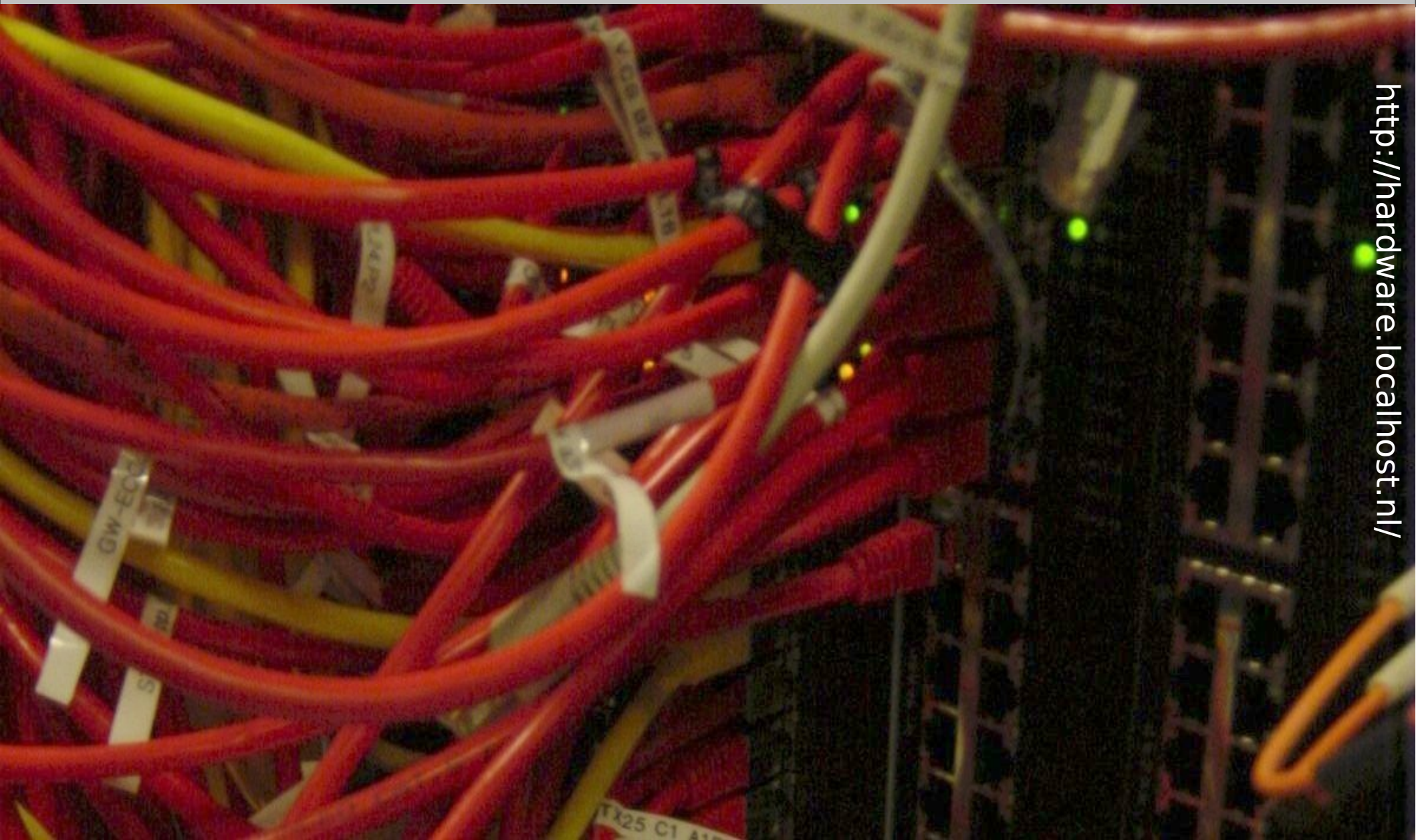
Brauche ich PA oder PI Adressen?
PA→ technischer Aufwand gering
PI → technischer Aufwand hoch

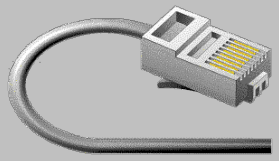
IP-Tapete


Colors:
assigned
RIPE
free
reserved
212.55.192.0/19 [7764 of 8192 IPs (94.8%) used]
212.55.192.0:
212.55.193.0:
212.55.194.0:
212.55.195.0:
212.55.196.0:
212.55.197.0:
212.55.198.0:
212.55.199.0:
212.55.200.0:
212.55.201.0:
212.55.202.0:
212.55.203.0:
212.55.204.0:
212.55.205.0:
212.55.206.0:
212.55.207.0:
212.55.208.0:
212.55.209.0:
212.55.210.0:
212.55.211.0:
212.55.212.0:
212.55.213.0:
212.55.214.0:
212.55.215.0:
212.55.216.0:
212.55.217.0:
212.55.218.0:
212.55.219.0:
212.55.220.0:
212.55.221.0:
212.55.222.0:
212.55.223.0:



Fragen ?

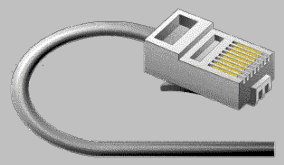




IP Pakete

Ziele:

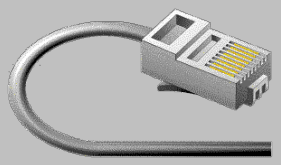
- IP Pakete
- ICMP Pakete
- TCP Pakete verstehen
- 3 Way Handshake
- Sliding Window
- UDP Pakete verstehen
- Well known Port Numbers kennen



IP Paket

	1Byte		2Byte	3Byte	4Byte
0	V	HL	Prio	Length	
4	ID			F	Frag
8	TTL		Proto	Checksm	
12	Src IP				
16	Dest IP				
20	Optional				PAD
	Data				

V	Version
HL	HeaderLength in Words
Prio	Priority Flags
Length	Länge des ganzen Flows
ID	Sequenznummer des Flows
F	Flags
Frag	Abstand des aktuellen Fragementes zum Anfang der Daten
TTL	Time to Live (Hops to Live)
Proto	Protokoll der Daten
Checksm	Header Checksumme
SrcIP	Source IP-Adresse
DestIP	Destination IP-Adresse
Optional	Optionale Header Daten
PAD	Füllbytes, da der IP-Header n*32Bit lang sein muss (→ HeaderLength)



IP Paket

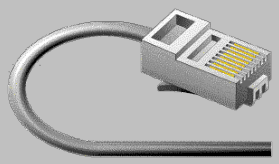
V Version (4Bit)
4 → IPv4

HL HeaderLength in Words (4Bit)
Die Länge des Headers in Worten (32Bit). Mögliche Werte sind 5 ... 16, entspricht einer Headerlänge von 20 bis 64Byte

Differentiated Services Field,Priority Flags
Bit 7... 2: Differentiated Services Codepoint
Bit 1: ECN-Cable Transport
Bit 0: ECN-CE

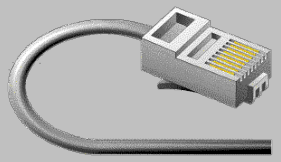
Length Länge des ganzen Pakets (16Bit)
Maximale Grösse eines IP-Paketes ist 2^{16} Byte

ID Identifikations Nummer des Flows
Jeder Flow hat eine eindeutige Nummer. Diese Nummer wird vom Absender vergeben.



IP Paket

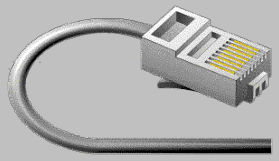
F	Flags Bit (7,6,5) Bit 7: Reserve Bit 6: Don't Fragment Bit 5: more Fragments
Frag	Abstand des aktuellen Fragments zum Anfang der Daten (13bit) * 8 Fragmente können daher nur ein mehrfaches von 8Byte betragen!
TTL	Time to Live (Hops to Live) Jeder Router, der das Paket verarbeitet verringert diesen Zähler. Wird der Wert 0 erreicht – und das Paket ist nicht am Ziel – wird das Paket verworfen und der Absender via einer ICMP-Meldung darüber informiert.
Proto	Protokoll der Daten IP-Protokoll Nummer, meistens (ICMP [1], TCP [6] oder UDP [17])



IP Paket

Checksm	Header Checksumme Quersumme über den ganzen Headerbereich
SrcIP	Source IP-Adresse
DestIP	Destination IP-Adresse
Optional	Optionale Header Daten Im Header können zusätzliche Felder eingefügt werden. Beispielsweise Authentication Header (RFC1826)…Mögliche Werte sind bei IANA [1] publiziert.
PAD	Füllbytes, da der IP-Header n*32Bit lang sein muss (→ HeaderLength)

[1] http://www.iana.org/assignments/ip-parameters

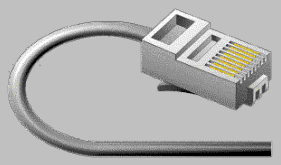


Fragmentierte Pakete

Ist das Paket Fragmentiert, so wird das Flag "More Fragments" bei allen Paketen – ausser dem letzten – gesetzt.

Anhand der Source-IP-Adresse und SequenzID können die Paket identifiziert werden.

Der Parameter FrameOffest gibt an an welcher Stelle das empfangene Paket eingesetzt werden muss.



Fragmentierte Pakete

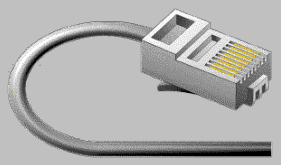
Total Length = 5008Byte

HeaderLength=20 Length=1500 MoreFragments=1 Offset=0	HeaderLength=20 Length=1500 MoreFragments=1 Offset=185	HeaderLength=20 Length=1500 MoreFragments=1 Offset=370	HL=20 Len=568 MF=0 Offset=555

Pro Paket können 1480Byte transportiert werden!

FragmentOffsets müssen mit 8 multipliziert werden, um den Platz im Datenstream zu 'finden'.

Siehe auch die Datei ip_fragment.libpcap

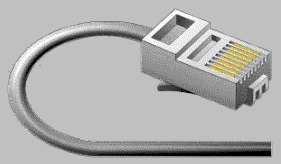


IP Paket

```
0000   00 0f 34 e7 8b ae 00 01   02 37 cc 95 81 00 00 03   ..4..... .7......
0010   08 00 45 08 00 72 11 42   40 00 3f 06 f7 9b d4 37   ..E..r.B @.?....7
0020   c4 4a d4 37 c5 e6 fd cf   0c ea 13 48 86 4d d6 86   .J.7.... ...H.M..
0030   a7 1b 80 18 ff ff 4c 53   00 00 01 01 08 0a 60 1d   ......LS ......`.
0040   a3 5e 15 43 0e df 3a 00   00 00 03 73 65 6c 65 63   .^.C..:. ...selec
0050   74 20 2a 20 66 72 6f 6d   20 4e 4f 5f 4f 46 5f 52   t * from  NO_OF_R
0060   41 54 49 4e 47 53 5f 50   45 52 5f 53 54 49 4d 55   ATINGS_P ER_STIMU
0070   4c 55 53 20 77 68 65 72   65 20 63 6f 75 6e 74 20   LUS wher e count
0080   3c 20 31 30                                        < 10
```

OSI 4, IP-Payload
OSI 3, IP-Header
OSI 1,2, Ethernet Header mit VLAN Tag

ACHTUNG: Der Ethertype vom Paket **muss** IP (0x0800) sein, sonst ist es KEIN IPv4 Paket!

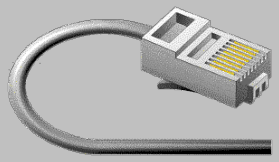


IP Paket im Detail

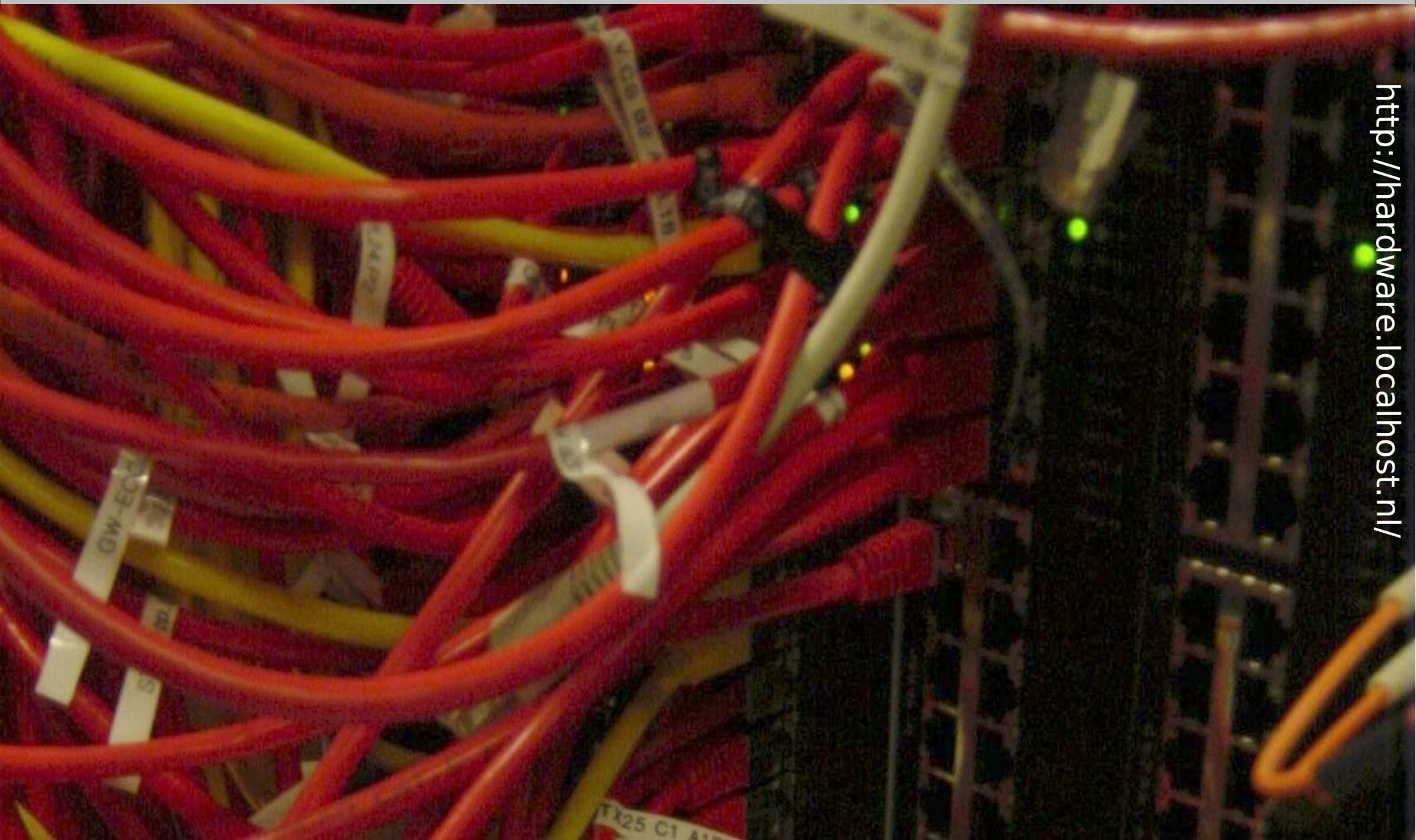
HexDump eines IP-Paketes

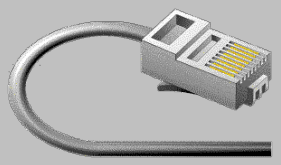
```
0000   00 0f 34 e7 8b ae 00 01   02 37 cc 95 81 00 00 03    ..4..... .7......
0010   08 00 45 08 00 72 11 42   40 00 3f 06 f7 9b d4 37    ..E..r.B @.?....7
0020   c4 4a d4 37 c5 e6 .. ..   .. .. .. .. .. .. .. ..    .J.7..
```

Offset	Wert	Bedeutung
0012:	45	Version 4, HeaderLength 5x4Byte = 20Byte
0013:	08	TOS/DSCP-Feld
0014:	0072	Total Length: 114
0016:	1142	ID: 4418
0018:	4000	Flag, FragOffset
001a:	3F	Time To Live
001b:	06	Protokoll: TCP
001c:	f79b	Header Checksum
001e:	d4.37.c4.4a	Source IP-Adresse (212.55.196.74)
0022:	d4.37.c5.e6	Destination IP-Adresse (212.55.197.230)



Fragen ?

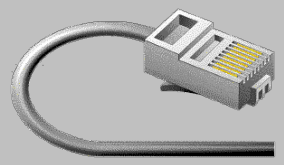




Internet Control Message Protocol, ICMP

ICMP Meldungen werden in verschiedenen Situationen versendet z.B. wenn ein Paket das Ziel nicht erreichen kann, wenn ein Router einen besseren Weg zu Ziel kennt, oder um die Erreichbarkeit eines Hosts zu testen.

Das Internet Protokoll (IP) ist so gebaut, dass der Ziel-Rechner mit grosser Wahrscheinlichkeit erreicht wird, eine Garantie, dass der Ziel Host erreicht werden kann, gibt es nicht!



Internet Control Message Protocol, ICMP

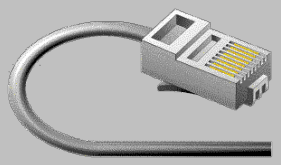
ICMP wird verwendet um dem Absender über Probleme bei der Übertragung zu informieren.

ICMP kann verwendet werden um Verbindungs-Probleme zu untersuchen.

Um endlose ICMP-Loops zu vermeiden werden nie ICMP-Meldungen aufgrund von ICMP-Paketen erzeugt!

Jede IP-Implementierungen muss ICMP unterstützen.

ICMP ist ein sehr wichtiger Teil vom Internet Protokoll und darf nicht blockiert werden.

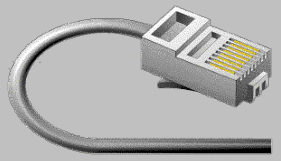


ICMP Paket

	1Byte	2Byte	3Byte	4Byte
	IP-Header ...			
0	Type	Code	Checksm	
4	ID		Sequenz#	
	Data			

ICMP-Paket basiert auf IP, darum muss vor dem ICMP-Paket ein IP-Header stehen!

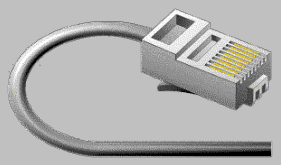
Type:	Art der ICMP Nachricht
Code:	Detaillierte Information zur Nachricht
Checksum:	Checksumme der ICMP Nachricht
ID:	Identifier
Sequenz#:	Sequenznummer
Data:	Weitere Daten (Meistens die Header vom Paket, das die Meldung verursachte, oder entsprechende Daten)



ICMP Type

ICMP Typen:

0	**Echo Replay**
1	Reserved
2	Reserved
3	**Destination unreachable**
4	**Source quench**
5	**Redirect**
6	Alternate Host Address
7	
8	**Echo Request**
9	Router Advertisement
10	Router solitation
11	**Time exceeded**
12	Parameter Problem
13	Timestamp request
14	Timestamp replay
15	Information request
16	Infromation Replay
17	Address mask request
18	Address mask replay
19	Reserved
20 – 29	Reserved
30	Traceroute
31	Conversation error
32	Mobile Host Redirect
33	IPv6 Where are You
34	IPv6 I Am Here
35	Mobile Registration Request
36	Mobile Registration Replay
37	Domain Name request
38	Domain Name replay
39	SKIP Algorithm Discovery Protokoll
40	Photuris, Security failures
41	Experimental mobility protokoll
42 – 255	Reserved

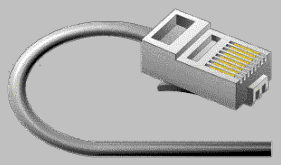


ICMP Paket

ICMP Pakete werden über IP versendet:

```
0000   00 01 02 37 cc 95 00 0f  34 e7 8b ae 08 00 45 c0   ...7.... 4.....E.
0010   00 38 54 08 00 00 ff 01  36 01 d4 37 c4 41 d4 37   .8T..... 6..7.A.7
0020   c4 4a 0b 00 f4 ff 00 00  00 00 45 a0 00 40 a1 49   .J...... ..E..@.I
0030   00 00 01 01 e5 33 d4 37  c4 4a d4 37 c5 e6 08 00   .....3.7 .J.7....
0040   dd f0 00 0f 1a 00                                 ......
```

Adresse / Offset	Wert	Bedeutung
0000 ... 0011:	Ethernet Header	EtherType muss 0x0800 sein!
0012 ... 0021:	IP Header	Protocoll muss ICMP sein!
0022:	0x0b	Type: Time Exceeded
0023:	0x00	Code
0024:	0xf4ff	Checksumme
0026:	0x0000	Identifier: 0
0028:	0x0000	Sequenznummer: 0
002a:	0x45a0..	Hier folgt eine Kopie des Paketes, das das "Problem" auslöste

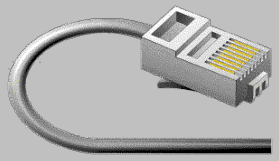


ICMP Echo Request

Wenn ein IP-Host eine **ICMP Echo Request** Anforderung bekommt, so antwortet er mit einer **ICMP Echo Replay** Meldung.

Damit kann getestet werden ob der Host IP mässig richtig konfiguriert ist.

ICMP Echo Requests können mit dem Befehl **`ping <IP-Adresse>`** erzeugt werden.

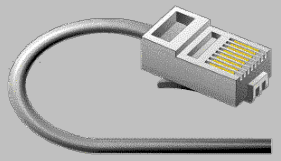


ICMP Destination unreachable

Dieser ICMP-Typ ist der wichtigste neben den Echo Request.

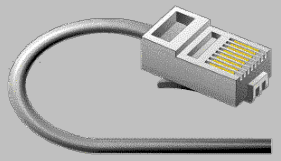
Destination unreachables werden von Routern auf dem Weg zum Ziel oder vom adressierten Host erzeugt.

Der Grund warum das Ziel nicht erreichbar ist wird im Code Feld der ICMP-Meldung genauer angegeben:



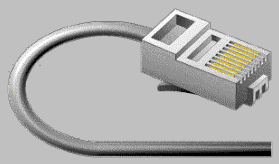
ICMP Destination unreachable

Code	Beschreibung
0	Network unreachable error
1	Host unreachable error
2	Protocol unreachable error. Das gewünschte Protokoll ist nicht unterstützt.
3	Port unreachable error. *Der gewünschte Port ist nicht erreichbar. In der Regel ist der Dienst hinter diesem Port nicht aktiv.*
4	The datagram is too big. *Das gesendete Paket ist zu gross*
5	Source route failed error.
6	Destination network unknown error.
7	Destination host unknown error.



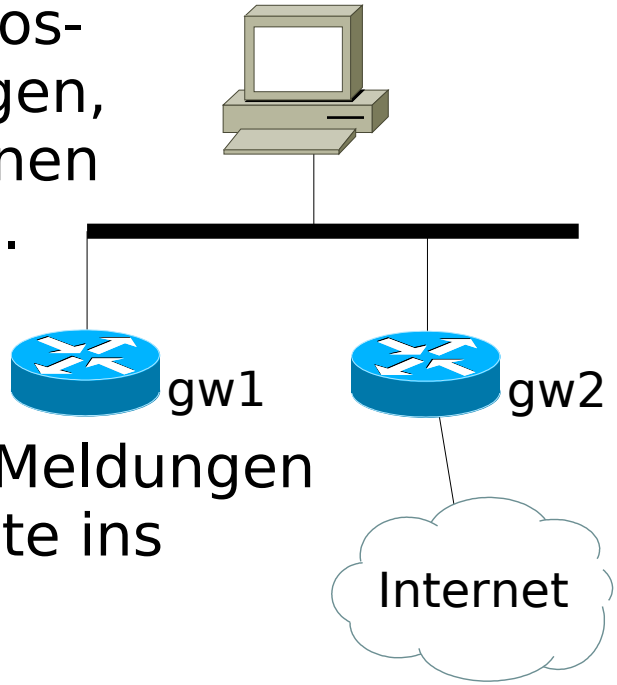
ICMP Destination unreachable

Code	Beschreibung
8	Source host isolated error. Obsolete.
9	The destination network is administratively prohibited. *Das Netzwerk ist gefiltert.*
10	The destination host is administratively prohibited. *Das Paket wurde vom Host ausgefiltert.*
11	The network is unreachable for Type Of Service.
12	The host is unreachable for Type Of Service.
13	Communication Administratively Prohibited. *Das Paket wurde auf einem Router gefiltert.*
14	Host precedence violation.
15	Precedence cutoff in effect.



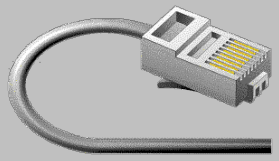
ICMP Redirect message

Router senden einem direkt angeschlossenen Rechner ICMP redirect Meldungen, wenn der Ziel-Adresse besser über einen anderen Router erreicht werden kann.



Wenn der Rechner, gw1 als Default-Gateway verwendet, so kann gw1 dem Rechner via einer ICMP redirect Meldungen mitteilen,dass er besser gw2 für Pakete ins Internet verwenden soll.

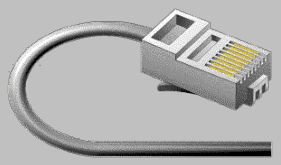
▶ ICMP-Redirect Meldungen sind problematisch, da jeder Host diese Nachrichten erzeugen kann und dadurch den Traffic entsprechend umgeleitet werden kann!
Redirect Meldungen sollen nur von lokalen Router versendet werden (Achtung IP-Address Spoofing!)!



ICMP Source quench message

Ein Router sendet dem Absender eine **ICMP Source quench Meldung**, wenn die Übertragungskapazität vom Router überlastet ist.

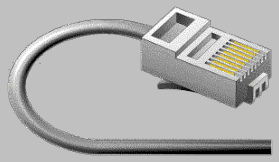
Weitere Pakete, die an diesen Router gesendet werden, können – mit sehr grosser Wahrscheinlichkeit – verworfen werden.



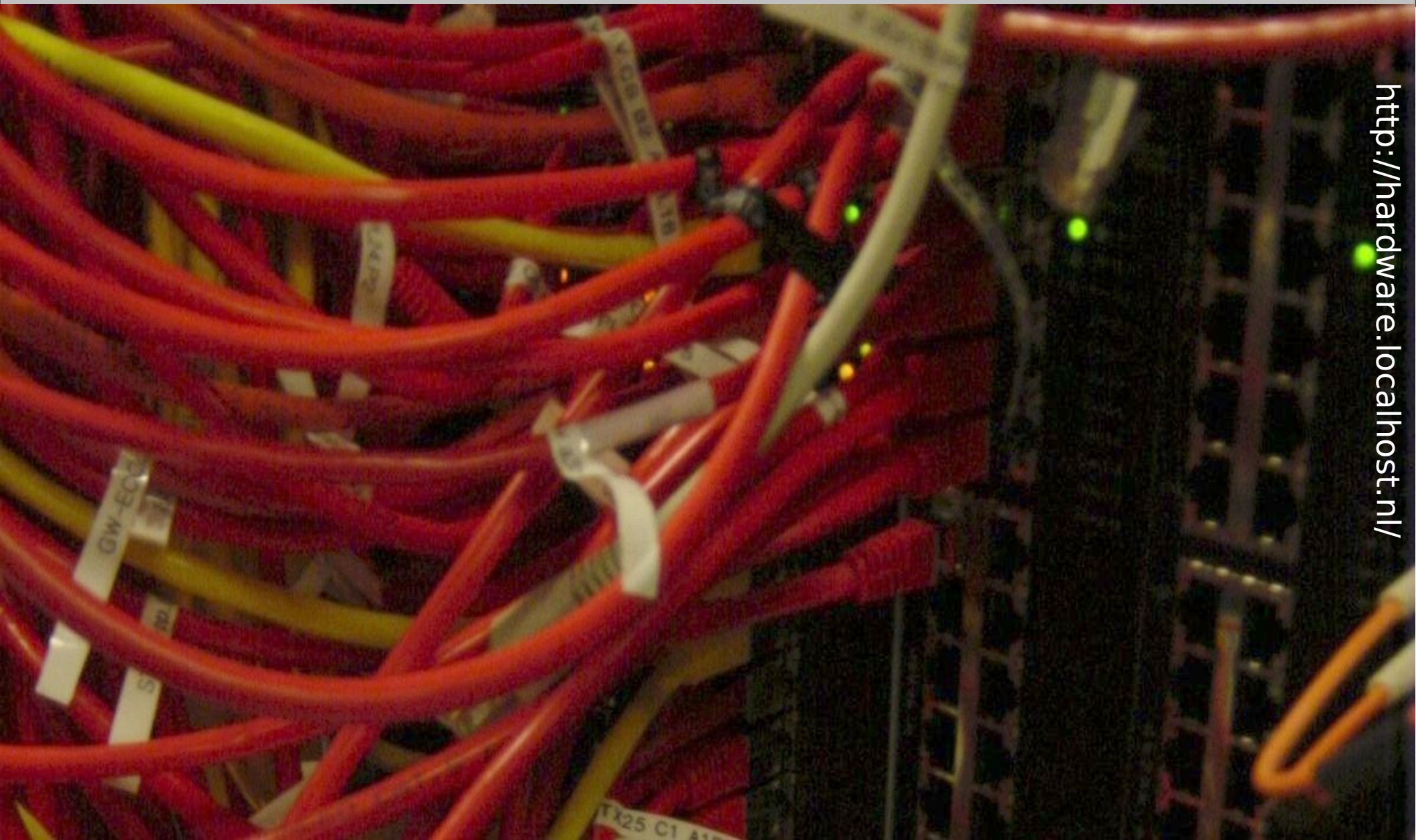
ICMP Time Exeeded

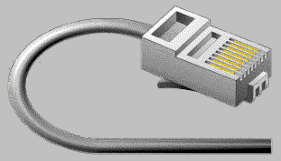
Es gibt 2 Time Exeeded Meldungen

- Wenn in einem IP-Paket das TTL Feld == 0 wird, wird das Paket verworfen und eine Time to Live exceeded ICMP-Meldung an den Absender gesendet. Durch das Time to Live Feld können Layer 3 Loops verhindert werden.
- Ein Host sendet eine ICMP Time Exceeded Meldungen wenn der Host nicht alle notwendigen Fragmente eines fragmentierten Paketes innerhalb eine bestimmten Zeit bekommen hat.
  Der Host verwirft die erhaltenen Fragmente.



Fragen ?





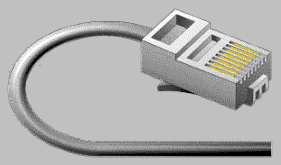
TCP / UDP Paket

TCP

- Verbindungsorientiert
- Zuverlässig
- Garantierte Reihenfolge der Daten
- Flexibilität in der Bandbreiten-Nutzung

UDP

- Verbingsloses Protokoll
- kleiner Overhead
- Schnell
- Reihenfolge der Daten ist **nicht** garantiert



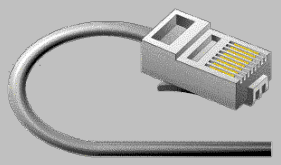
Transmission Control Protocol (TCP)

TCP ist verbindungsorientiert.

Das bedeutet, das bevor Daten ausgetauscht werden können, eine Verbindung aufgebaut werden muss.

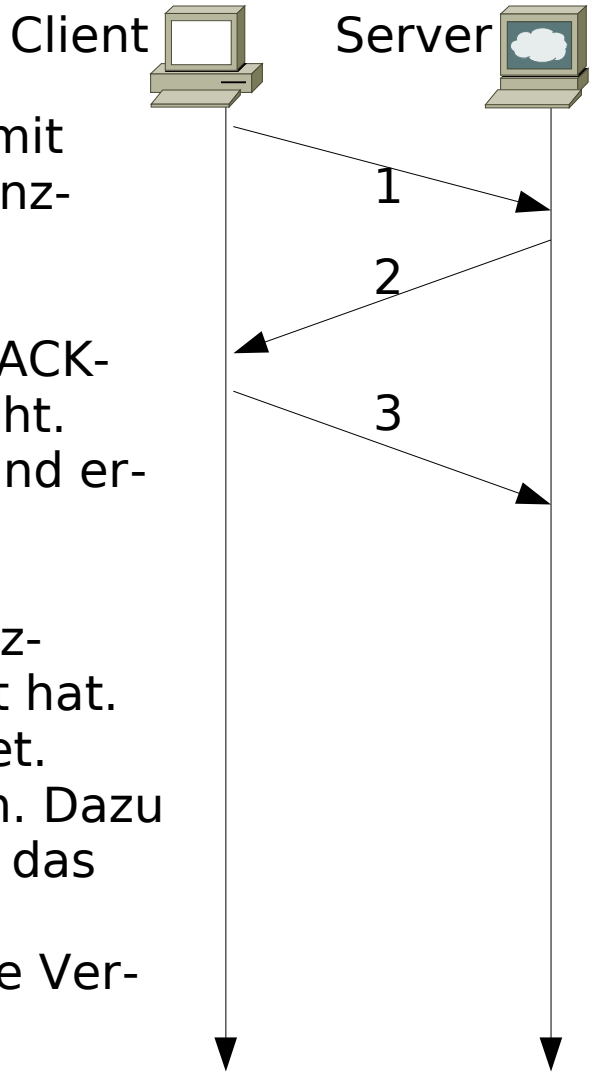
Der Verbindungsaufbau erfolgt in 'Three-way-Handshake'

Eingesetzt wird TCP wo garantiert sein muss, dass die Daten ankommen und die Reihenfolge der Daten wichtig ist:
Web (http), Mail (smtp, pop3, imap4), Telnet, ssh, ...



Three way Handshake

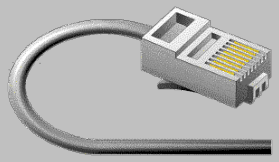
Verbindungsaufbau:



1) Der Client sendet dem Server eine TCP Paket mit dem SYN-Bit gesetzt, und einer beliebigen Sequenz-Nummer X.

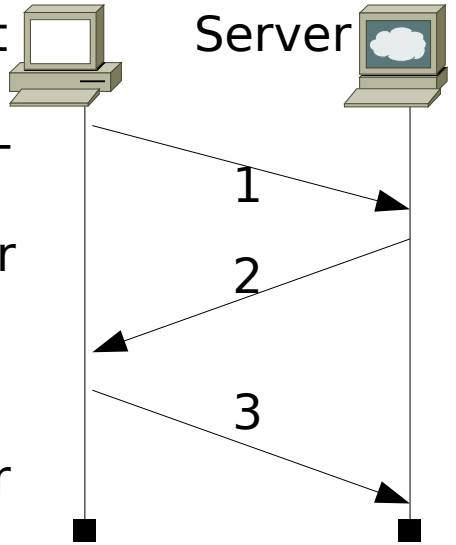
2) Der Server bestätigt das Paket, in dem er das ACK-Flag setzt und die Sequenz-Nummer X um 1 erhöht. Gleichzeitig setzt der Server auch das SYN-Flag und erzeugt eine eigene Sequenz-Nummer Y

3) Der Client sieht, dass der Server seine Sequenz-Nummer X bekommen hat und richtig verarbeitet hat. Die Verbindung Client -> Server ist damit geöffnet.
Der Client muss das Paket vom Server bestätigen. Dazu erhöht er die Sequenz-Nummer Y um 1 und setzt das ACK-Flag.
Wenn der Server das Paket bekommt, ist auch die Verbindung Server -> Client geöffnet



Three way Handshake

Verbindunsabbau

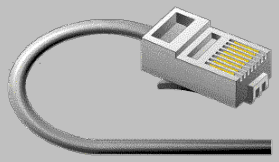


1) Wenn ein Verbindungspartner die Verbindung abbauen will, so sendet er ein Pakt mit der richtigen Sequenz-Nummer und dem gesetzten FIN-Flag. (hier im Beispiel initiiert der Client den Abbau)

2) Das Paket wird mit einem gesetzten ACK-Flag vom Server bestätigt. Gleichzeitig verlangt auch der Server, dass die Verbindung terminiert wird, in dem er das FIN-Flag setzt.

3) Diese Paket wird vom Client mit einem ACK Paket bestätigt.

Wichtig: Der Verbindungsabbau kann auch vom Server initiiert werden! (Es muss nur ein Paket mit der richtigen Sequenz-Nummer und den entsprechenden Flags gesendet werde)



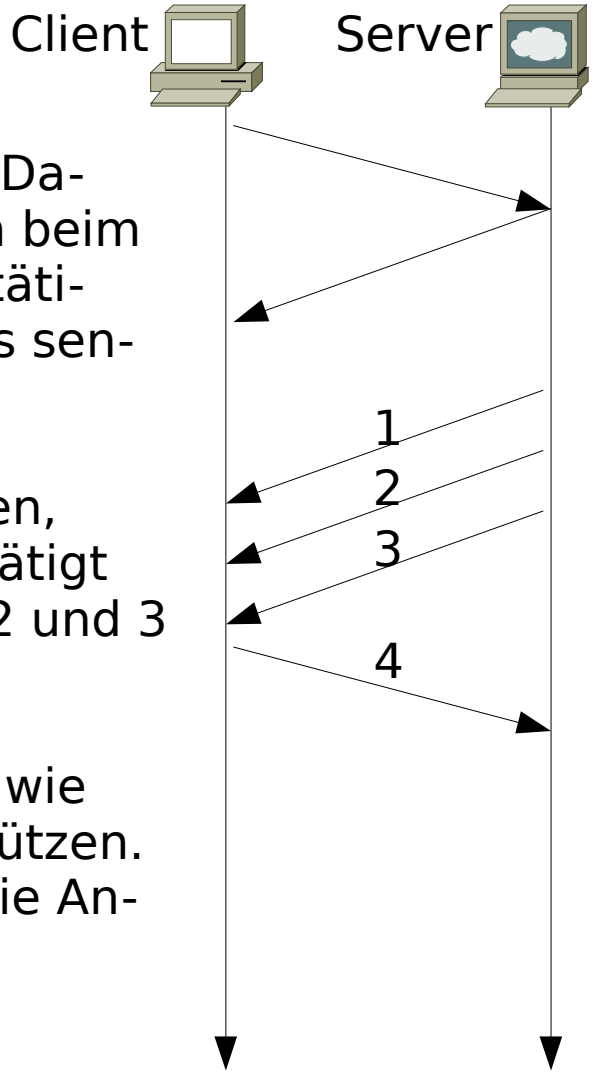
Datentransfer

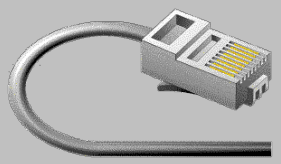
Datentransfer:

In der Regel wird jedes Datenpaket bestätigt. Dadurch wird sichergestellt, dass die Daten auch beim Empfänger angekommen sind. Fehlt eine Bestätigung, so muss der Sender das Paket nochmals senden

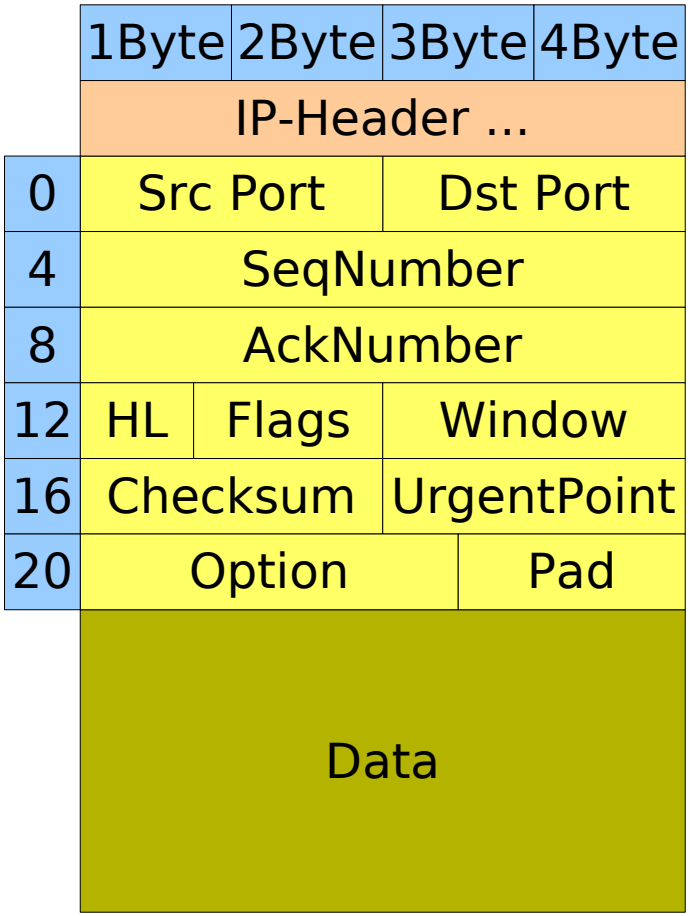
Um die Anzahl der Bestätigungen zu reduzieren, können auch mehrere Paket miteinander bestätigt werden. Mit dem Paket 4 werden die Paket 1,2 und 3 miteinander bestätigt.

Die Kommunikationspartner sprechen sich ab wie viele ausstehenden Bestätigungen sie unterstützen. Je nach Auslastung vom Netzwerk kann sich die Anzahl angepasst werden. (**sliding window**)



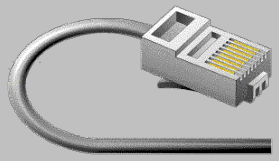


TCP Paket



TCP-Paket basiert auf IP, darum muss vor dem TCP-Paket ein IP-Header stehen!

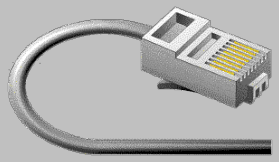
SrcPort:	Source Port
Dst Port:	Destination Port
SeqNumber:	eigene Sequenz-Nummer
AckNumber:	Acknowlage Sequenz-Nummer
HL:	4Bit Header Length in 32bit Worten
Flags:	12Bit Verschiedene Flags
Window:	Windowsize in 32Byte 'Paketen'
Checksum:	Checksumme der TCP Nachricht
UrgentPoint:	Dieser Pointer zeigt auf das Ende der Dringenden Daten hin.
Optionen:	Optionale TCP Header
Pad:	Fülldaten, damit der Header die spezifizierte Headerlänge bekommt.



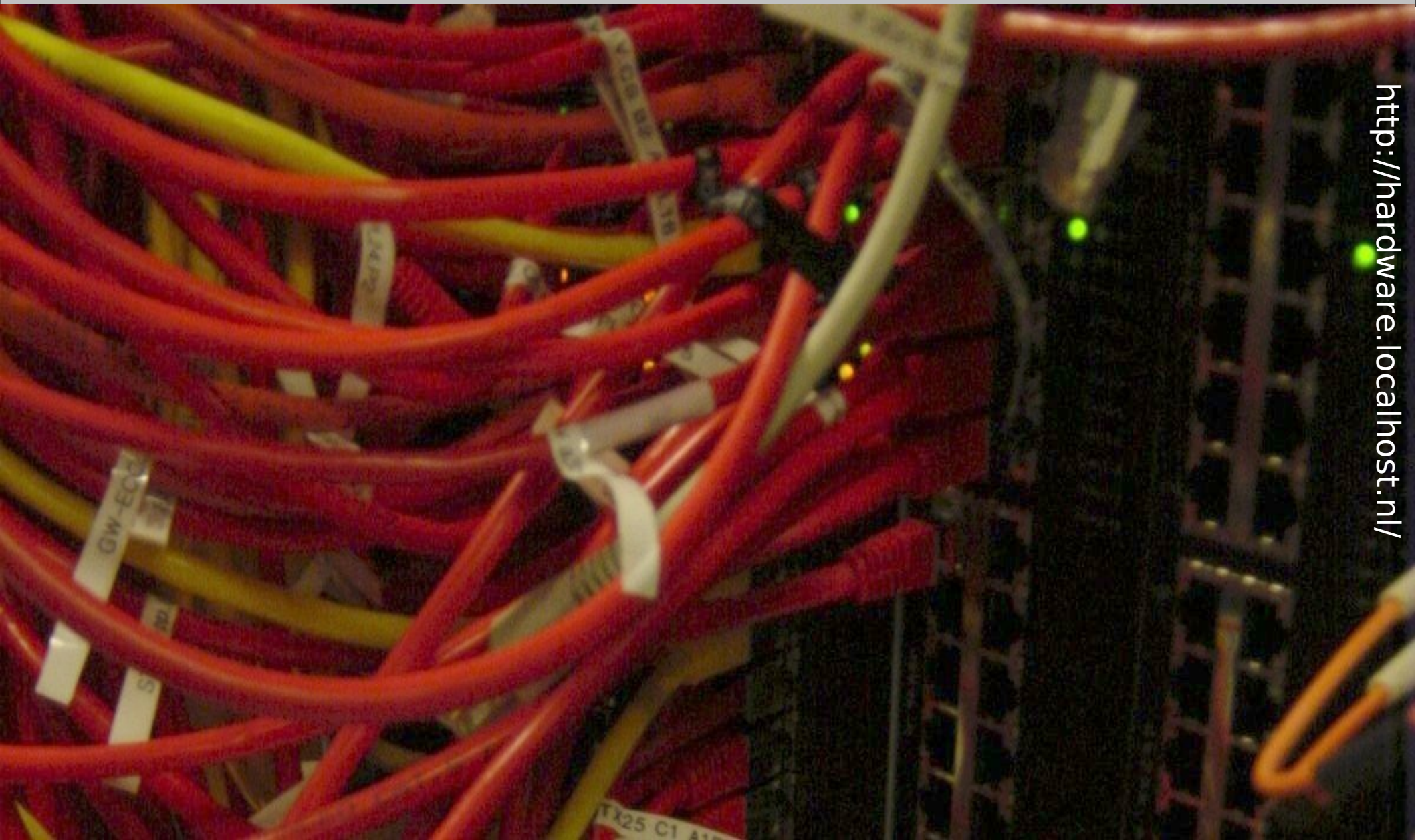
TCP Flags

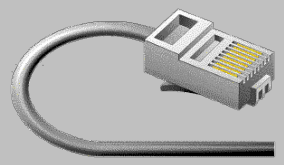
Im Moment sind folgende TCP Flags (≠IP Flags) definiert:

Fin:	Verlangt, dass die Verbindung beendet wird
Syn:	Aufbau ein Verbindung
Reset:	Zurücksetzen einer Verbindung
Push:	Die Daten sollen sofort verarbeitet werden ohne dass diese gepuffert werden.
Acknowledgment:	Der Header enthält eine gültige ACK-Sequenz-Nummer
Urgent:	Der UrgentPointer enthält eine gültige Angabe
ECN-ECHO:	verwendet für ECN
Congestion Window Reduce:	verwendet für ECN



Fragen ?



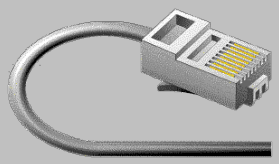


User Datagram Protocol (UDP)

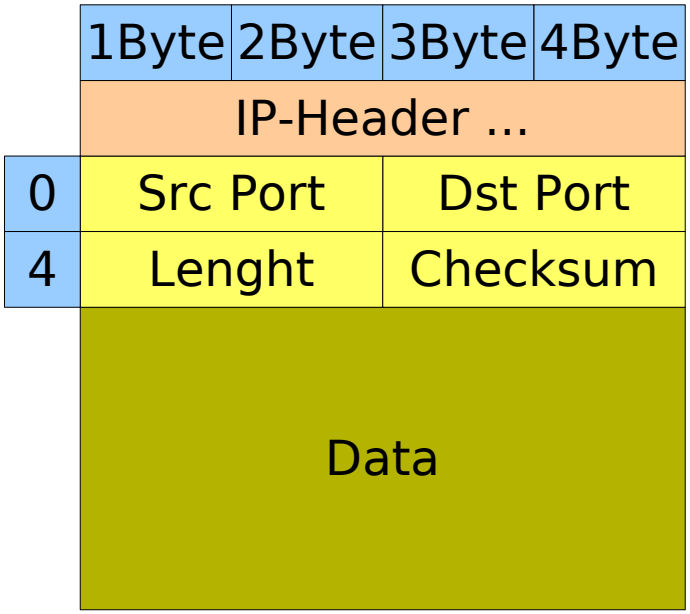
UDP

Verbingsloses Protokoll
kleiner Overhead
Schnell
Vermeiden redundanten Transportkontrollen

Eingesetzt wird UDP vor allem für Dienste die Pakete ohne grossen Protokoll overhead versenden müssen:
Namensauflösung (DNS), DateiSysteme (NFS, SMB, ...), VoIP, BOOTP, DHCP, SNMP

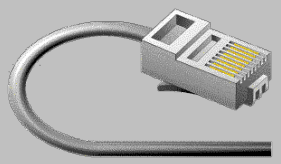


UDP Paket



UDP-Paket basiert auf IP, darum muss vor dem UDP-Paket ein IP-Header stehen!

SrcPort:	Source Port
Dst Port:	Destination Port
Length:	Länge des UDP Paketes
Checksum:	Checksumme der UDP Nachricht

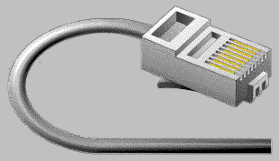


UDP Paket

UDP Pakete werden über IP versendet:

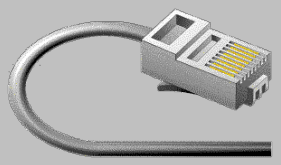
```
0000   00 0f 34 e7 8b ae 00 01   02 37 cc 95 08 00 45 00   ..4..... .7....E.
0010   00 3b 00 00 40 00 40 11   e3 2e d4 37 c4 4a c1 f6   .;..@.@. ...7.J..
0020   fd 0a 83 71 00 35 00 27   cc 68 1c 73 01 00 00 01   ...q.5.' .h.s....
0030   00 00 00 00 00 00 03 77   77 77 06 73 77 69 74 63   .......w ww.switc
0040   68 02 63 68 00 00 01 00   01                       h.ch.... .
```

Adresse / Offset	Wert	Bedeutung
0000 ... 000d:		Ethernet Header Protokoll muss IP sein!
000e ... 0021:		IP Header Protokoll muss UDP sein!
0022:	0x8371	SRC Port: 33649
0024:	0x0035	DST Port: 53 (Domain)
0026:	0x0027	Länge 39Byte (inkl. UDP Header)
0028:	0xcc68	Checksumme
002a ... 0049:	0x1c ...	Nutzdaten



UDP, TCP Portnummern

- UDP und TCP verwenden Portnummern um den Service zu adressieren.
- Bekannte Services haben fixe Portnummern (well known port number), welche von der IANA verwaltet werden.
- Ein Service kann – muss aber nicht – über beide Protokolle UDP oder TCP implementiert werden:
- DNS funktioniert sowohl über UDP als auch TCP.
- HTTP funktioniert nur über TCP.
- DHCP funktioniert nur als UDP, ...

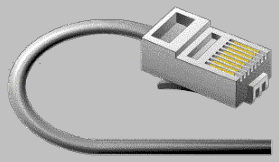


Portnummern, die man kennt

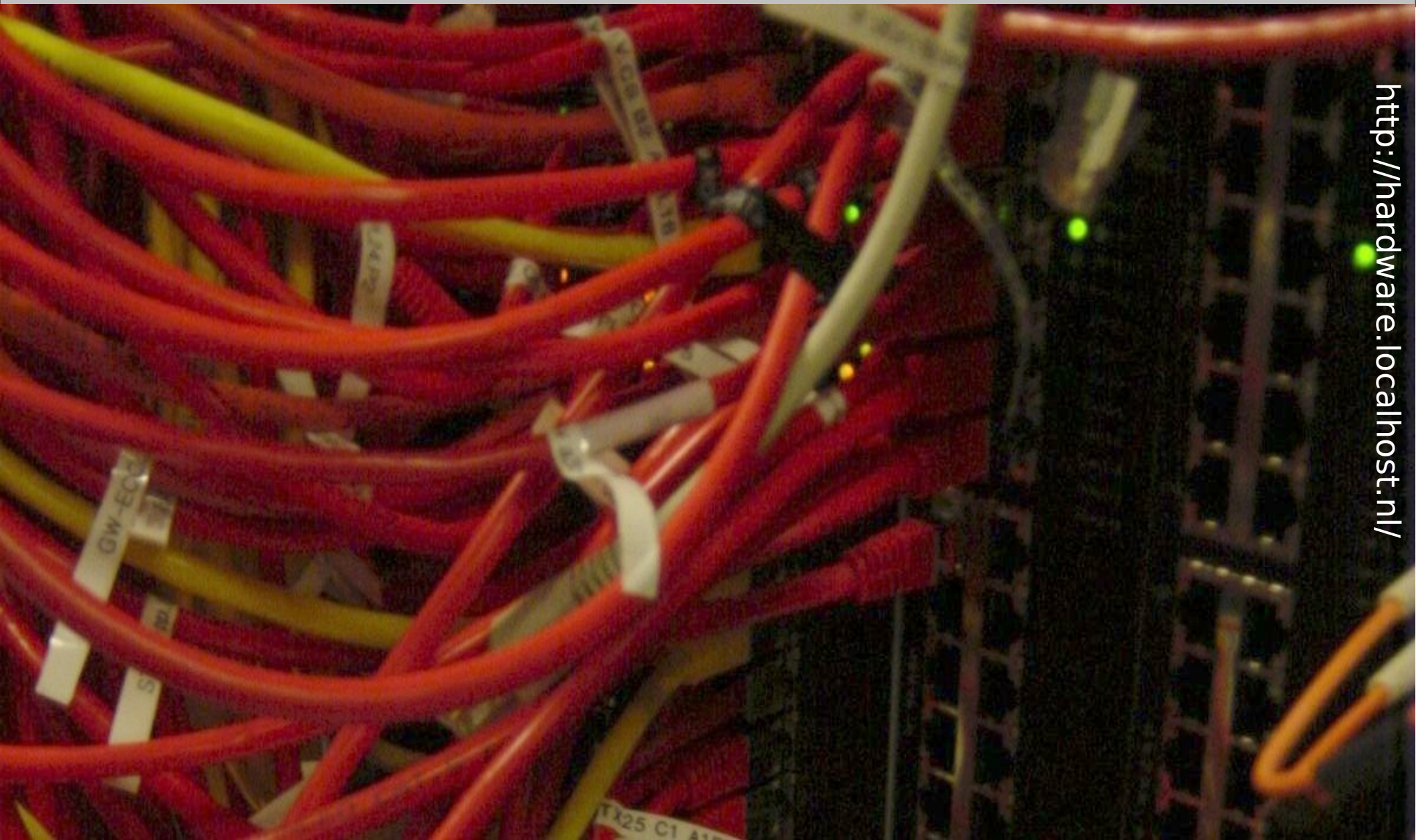
Einige Portnummern sind so verbreitet, dass man diese kennen sollte:

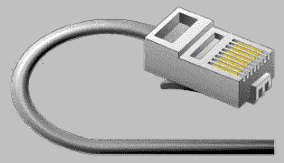
```
ftp-data 20/tcp
ftp      21/tcp
ssh      22/tcp
telnet   23/tcp
smtp     25/tcp
domain   53/udp  # dns
domain   53/tcp  # dns
bootps   67/udp  # dhcp
bootpc   68/udp  # dhcp
www      80/tcp
pop3     110/tcp
sunrpc   111/udp portmapper
sunrpc   111/tcp portmapper
imap     143/tcp
snmp     161/udp
snmp-trap        162/udp
```

In der Datei `/etc/services` sind die Portnummern aufgelistet.



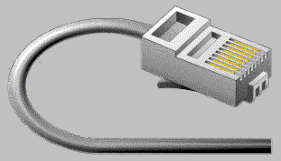
Fragen ?





IPv6

2001:db8:30:11:201:2ff:fe37:cc95/64



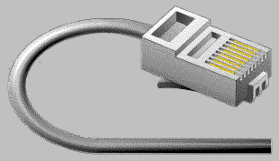
IPv6

Der Adressraum ist bei IPv4 ist zu klein. Es sind maximal 2^32 (~4.2Mia) Adressen möglich.
(Damit ist es nicht möglich, dass jeder Mensch eine IP-verwende kann!)

IPv6 erweitert den Adressraum auf 2^128 mögliche Adressen.
340'282'366'920'938'463'463'374'607'431'768'211'456

340 Sextillionen 282 Quintilliarden 366 Quintillionen 920 Quadrilliarden 938 Quadrillionen 463 Trilliarden 463 Trillionen 374 Billiarden 607 Billionen 431 Milliarden 768 Millionen 211 Tausend und 456

Die Felder im IPv6-Header wurden gegenüber dem IPv4-Header vereinfacht.



IPv6 Notation

jeweils 4 Bit werden als HexZahl (0-9a-f) geschrieben
4 dieser HexZahlen werden gruppiert und mittels Doppelpunkten getrennt

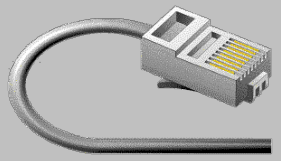
2001:0db8:0000:1234:0000:0000:0000:0001

führende Nullen können weggelassen werden:

2001:db8:0:1234:0:0:0:1

eine einzige Sequenz von :0:0: kann durch :: ersetzt werden

2001:db8:0:1234::1 nicht aber ~~2001:db8::1234::1~~

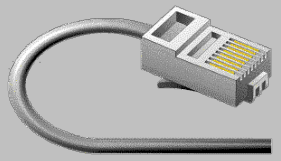


IPv6 Netzmasken

Es gibt keine Netzklassen wie bei IPv4
Die Netzmaske wird immer in der Slash-Notation geschrieben

2001:db8:0:1234::1**/64**

Gebräuchlich sind folgende Netzmasken
/128 eine einzelne IPv6 Adresse
/64 Ein einzelnes IPv6 Netz
/48 Mehrere Subnetze (65536 Netze)
/32 Kleinstes Netz, das von den RIRs an Provider vergeben wird

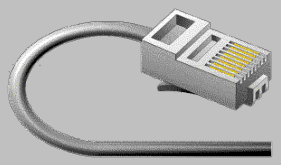


IPv6 / IPv4

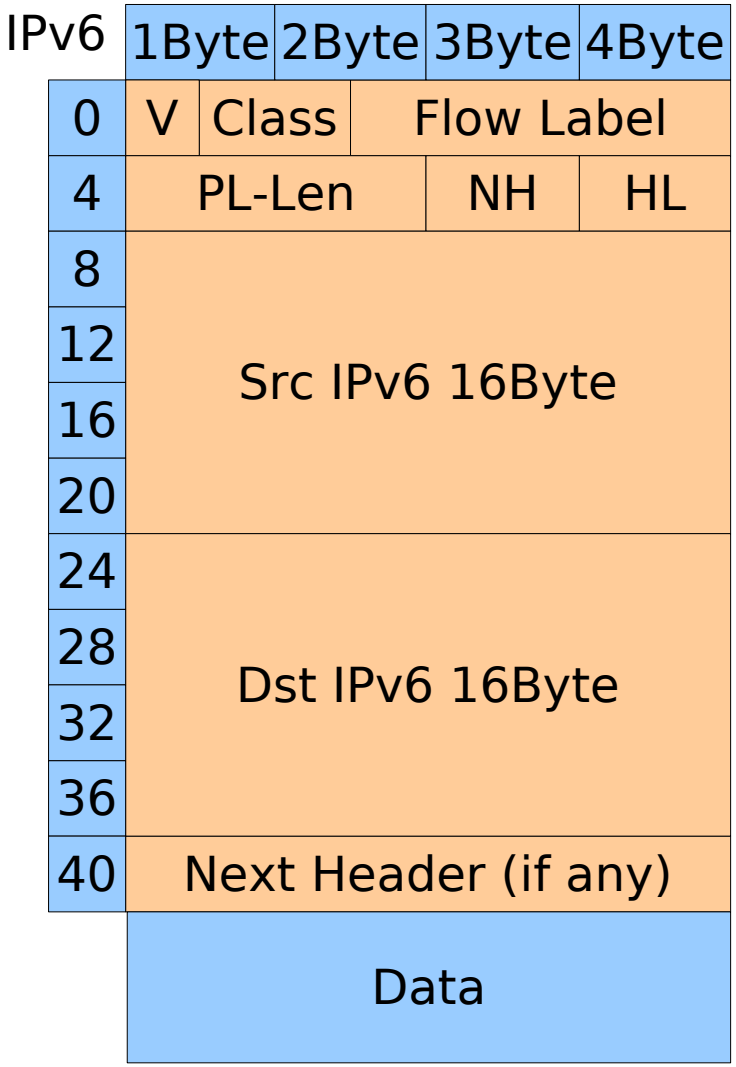
IPv6 und IPv4 können gleichzeitig verwendet werden:

```
heuer$ host heuer.org
heuer.org has address 212.55.197.226
heuer.org has IPv6 address 2001:8a8:30:10::226
```

Kennt der Rechner beide Protokolle (IPv6 und IPv4) kennt, so wird IPv6 bevorzugt.



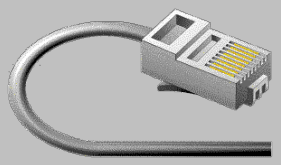
IPv6 Header



Die Felder im IPv6-Header wurden gegenüber dem IPv4-Header vereinfacht:

V	Version 6
Class	traffic Class
Flow Label	Flow Label (Id)
PL-Len	Payload Length
NH	Zeiger zum Next Header
HL	Hop Limit
Src Addr	Source Adresse
Dst Addr	Destination Adresse
N Header	Weitere Header wenn vorhanden

Im Paket kann die IPv6 Adresse nicht verkürzt eingetragen werden!



IPv4 / IPv6 Header

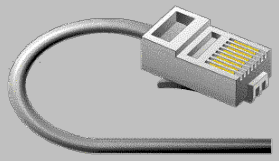
Vergleich vom IPv4 und IPv6 Header (ohne Headeroptionen)

IPv4

	1Byte		2Byte	3Byte	4Byte
0	V	HL	Prio	Length	
4	ID			F	Frag
8	TTL		Proto	Checksm	
12	Src IP				
16	Dst IP				
20	Data				

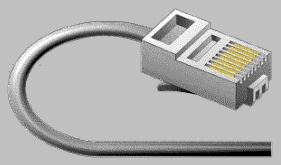
IPv6

	1Byte	2Byte	3Byte	4Byte
0	V	Class	Flow Label	
4	PL-Len		NH	HL
8	Src IPv6 16Byte			
12				
16				
20				
24	Dst IPv6 16Byte			
28				
32				
36				
40	Data			



Header: IPv4 vs IPv6

IPv6	Beschreibung	IPv4
Version	Version	Version
Class	traffic Class	~ Prio
Flow Label	Flow Label (Id)	ID/Sequenz
PL-Len	Payload Length	~ Length
NH	Zeiger zum Next Header	~ HeaderLength / Proto / Optional Header
HL	Hop Limit	TTL
Src Addr	Source Adresse	Src Addr
Dst Addr	Destination Adresse	Dst Addr
N Header	Weitere Header	n/a
n/a		Checksumme, Flags, Fragments

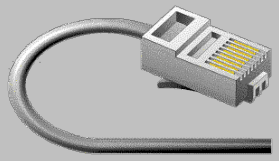


IPv6 Header

57 12.985048 2001:8a8:20::26 2001:8a8:30:11::2 SSH Encrypted response packet l

- Frame 57 (166 bytes on wire, 166 bytes captured)
- Ethernet II, Src: Cisco_e7:8b:ae (00:0f:34:e7:8b:ae), Dst: 3com_37:cc:95 (00:01:02:37:cc
- 802.1Q Virtual LAN
- Internet Protocol Version 6
  - Version: 6
  - Traffic class: 0x00
  - Flowlabel: 0x00000
  - Payload length: 108
  - Next header: TCP (0x06)
  - Hop limit: 61
  - Source address: 2001:8a8:20::26 (2001:8a8:20::26)
  - Destination address: 2001:8a8:30:11::2 (2001:8a8:30:11::2)
- Transmission Control Protocol, Src Port: ssh (22), Dst Port: 45569 (45569), Seq: 2036, A
- SSH Protocol

```
0010  86 dd 60 00 00 00 00 6c  06 3d 20 01 08 a8 00 20   ..`....l .= .... 
0020  00 00 00 00 00 00 00 00  00 26 20 01 08 a8 00 30   ........ .& ....0
0030  00 11 00 00 00 00 00 00  00 02 00 16 b2 01 67 17   ........ ......g.
0040  c9 12 c8 55 e1 42 80 18  00 5a 1d dc 00 00 01 01   ...U.B.. .Z......
0050  08 0a 84 f2 06 7a 31 d5  ad 28 92 b4 d7 24 c5 14   .....z1. .(...$..
0060  e7 f9 07 de 05 e3 94 93  68 82 b0 e5 e2 71 ee ac   ........ h....q..
0070  db 3a 58 24 a3 e3 df a0  6c 13 14 98 3a de eb 92   .:X$.... l...:...
0080  58 c8 8c 91 f0 00 40 41  36 0a a0 bd 55 59 8c dd   X.....@A 6...UY..
0090  b6 f0 f3 eb 48 4a 02 82  3b 91 17 f2 a4 d9 aa 33   ....HJ.. ;......3
00a0  1a 4c 77 67 15 c8                                  .Lwg..
```



IPv6

Layer 4 Protokolle sind unverändert

IPv6 kennt folgende Adressarten:

- Unicast
- Multicast
- Link Local

IPv6 kennt keine Broadcast Adresse!