Internet Protocol

Christian Zahler

5 Das Internet Protocol Version 4

Hauptaufgaben des IP-Protokolls:

- **Adressierung** von Netzknoten
- **Routing** (Wegesuche im Netz)
- **Zerlegung** des Datenstroms in **Pakete**; ein **IP-Datenpaket** kann **maximal 65536 Byte** groß sein.

Jeder Rechner auf der ganzen Welt braucht eine eindeutige Adresse, um im Internet oder in einem lokalen TCP/IP-Netzwerk erkannt zu werden, die so genannte IP-Adresse. In der derzeit gültigen Version 4 des Internet Protokolls ist die IP-Adresse eine 32-stellige Binärzahl, also etwa:

```
11011001.01010011.11001111.00010001
```

Meist fasst man 8 Binärstellen *(bits)* zu einem Byte zusammen, dessen dezimalen Wert man berechnet. Die "Kurzschreibweise" *(dotted decimal)* der oben angeführten IP-Adresse würde daher zum Beispiel lauten:

```
217.83.207.17
```

5.1 Zuweisung von IP-Adressen

IP-Adressen können auf zwei Arten vergeben werden:

- **Statische Konfiguration**: Die IP-Konfiguration wird manuell festgelegt und ändert sich nicht; in Windows wird die Konfiguration in den Netzwerkeigenschaften (Systemsteuerung) festgelegt.
- **Dynamische Konfiguration**: Die IP-Konfiguration wird von einem DHCP-Server *(Dynamic Host Configuration Protocol)* bezogen; die konkrete IP-Adresse wird bei jedem Neustart vom DHCP-Server neu zugewiesen und kann sich daher auch ändern.

5.2 Vergabe von IP-Adressen

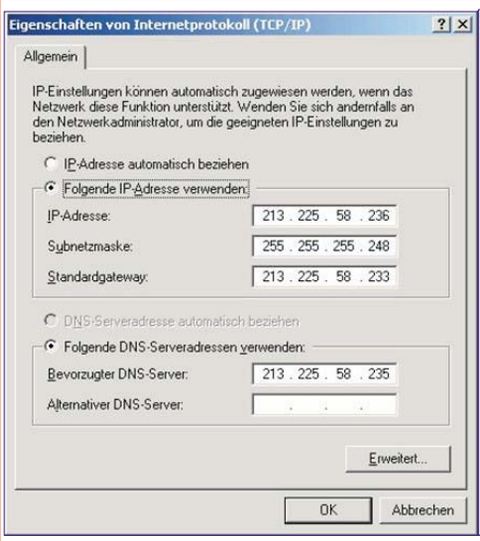
Man unterscheidet:

- **Öffentliche IP-Adressen (Public IPs)**: Diese Adressen werden von der *Internet Number Association (IANA)* vergeben. Diese Adressbereiche sind weltweit eindeutig und werden zur Adressierung von Geräten verwendet, die im Internet erreicht werden sollen. Solche Adressen können Sie über Ihren Internet Service Provider beziehen (nicht direkt bei der IANA).

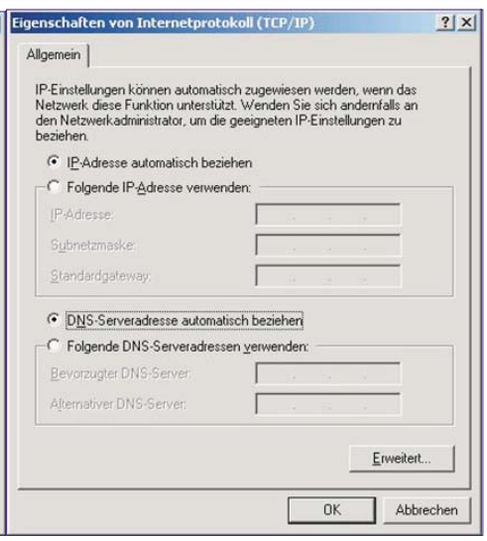
Quelle: *http://www.iana.org*

```
INTERNET PROTOCOL V4 ADDRESS SPACE
(last updated 03 August 2004) The allocation of Internet Protocol
version 4 (IPv4) address space to various registries is listed here.
Originally, all the IPv4 address spaces was managed directly by the
IANA. Later parts of the address space were allocated to various ot-
her registries to manage for particular purposes or regional areas
of the world.  RFC 1466 [RFC1466] documents most of these alloca-
tions.
Address
Block   Date    Registry - Purpose              Notes or Reference
-----   ------  --------------------------      ------------------
000/8   Sep 81   IANA - Reserved
001/8   Sep 81   IANA - Reserved
002/8   Sep 81   IANA - Reserved
003/8   May 94   General Electric Company
004/8   Dec 92   Bolt Beranek and Newman Inc.
005/8   Jul 95   IANA - Reserved
006/8   Feb 94   Army Information Systems Center
007/8   Apr 95   IANA - Reserved
008/8   Dec 92   Bolt Beranek and Newman Inc.
009/8   Aug 92   IBM
010/8   Jun 95   IANA - Private Use             See [RFC1918]
011/8   May 93   DoD Intel Information Systems
012/8   Jun 95   AT&T Bell Laboratories
013/8   Sep 91   Xerox Corporation
014/8   Jun 91   IANA - Public Data Network
015/8   Jul 94   Hewlett-Packard Company
016/8   Nov 94   Digital Equipment Corporation
017/8   Jul 92   Apple Computer Inc.
018/8   Jan 94   MIT
019/8   May 95   Ford Motor Company
020/8   Oct 94   Computer Sciences Corporation
021/8   Jul 91   DDN-RVN
022/8   May 93   Defense Information Systems Agency
023/8   Jul 95   IANA - Reserved
024/8   May 01   ARIN - Cable Block (Formerly IANA - Jul 95)
025/8   Jan 95   Royal Signals and Radar Establishment
026/8   May 95   Defense Information Systems Agency
027/8   Apr 95   IANA - Reserved
028/8   Jul 92   DSI-North
029/8   Jul 91   Defense Information Systems Agency
030/8   Jul 91   Defense Information Systems Agency
031/8   Apr 99   IANA - Reserved
032/8   Jun 94   Norsk Informasjonsteknology
033/8   Jan 91   DLA Systems Automation Center
034/8   Mar 93   Halliburton Company
035/8   Apr 94   MERIT Computer Network
036/8   Jul 00   IANA-Reserved (Formerly Stanford Univ. - Apr 93)
037/8   Apr 95   IANA - Reserved
038/8   Sep 94   Performance Systems International
039/8   Apr 95   IANA - Reserved
040/8   Jun 94   Eli Lily and Company
041/8   May 95   IANA - Reserved
042/8   Jul 95   IANA - Reserved
043/8   Jan 91   Japan Inet
044/8   Jul 92   Amateur Radio Digital Communications
045/8   Jan 95   Interop Show Network
046/8   Dec 92   Bolt Beranek and Newman Inc.
047/8   Jan 91   Bell-Northern Research
048/8   May 95   Prudential Securities Inc.
049/8   May 94   Joint Technical Command  (Returned to IANA  Mar 98)
050/8   May 94   Joint Technical Command  (Returned to IANA  Mar 98)
051/8   Aug 94   Deparment of Social Security of UK
052/8   Dec 91   E.I. duPont de Nemours and Co., Inc.
053/8   Oct 93   Cap Debis CCS
054/8   Mar 92   Merck and Co., Inc.
055/8   Apr 95   Boeing Computer Services
056/8   Jun 94   U.S. Postal Service
057/8   May 95   SITA
058/8   Apr 04   APNIC                          (whois.apnic.net)
059/8   Apr 04   APNIC                          (whois.apnic.net)
060/8   Apr 03   APNIC                          (whois.apnic.net)
061/8   Apr 97   APNIC                          (whois.apnic.net)
062/8   Apr 97   RIPE NCC                       (whois.ripe.net)
063/8   Apr 97   ARIN                           (whois.arin.net)
064/8   Jul 99   ARIN                           (whois.arin.net)
065/8   Jul 00   ARIN                           (whois.arin.net)
066/8   Jul 00   ARIN                           (whois.arin.net)
067/8   May 01   ARIN                           (whois.arin.net)
068/8   Jun 01   ARIN                           (whois.arin.net)
069/8   Aug 02   ARIN                           (whois.arin.net)
070/8   Jan 04   ARIN                           (whois.arin.net)
071/8   Aug 04   ARIN                           (whois.arin.net)
072/8   Aug 04   ARIN                           (whois.arin.net)
073/8   Sep 81   IANA - Reserved
074/8   Sep 81   IANA - Reserved
075/8   Sep 81   IANA - Reserved
076/8   Sep 81   IANA - Reserved
077/8   Sep 81   IANA - Reserved
078/8   Sep 81   IANA - Reserved
079/8   Sep 81   IANA - Reserved
080/8   Apr 01   RIPE NCC                       (whois.ripe.net)
081/8   Apr 01   RIPE NCC                       (whois.ripe.net)
082/8   Nov 02   RIPE NCC                       (whois.ripe.net)
083/8   Nov 03   RIPE NCC                       (whois.ripe.net)
084/8   Nov 03   RIPE NCC                       (whois.ripe.net)
085/8   Apr 04   RIPE NCC                       (whois.ripe.net)
086/8   Apr 04   RIPE NCC                       (whois.ripe.net)
087/8   Apr 04   RIPE NCC                       (whois.ripe.net)
088/8   Apr 04   RIPE NCC                       (whois.ripe.net)
089/8   Sep 81   IANA - Reserved
090/8   Sep 81   IANA - Reserved
091/8   Sep 81   IANA - Reserved
092/8   Sep 81   IANA - Reserved
093/8   Sep 81   IANA - Reserved
094/8   Sep 81   IANA - Reserved
095/8   Sep 81   IANA - Reserved
096/8   Sep 81   IANA - Reserved
097/8   Sep 81   IANA - Reserved
098/8   Sep 81   IANA - Reserved
099/8   Sep 81   IANA - Reserved
100/8   Sep 81   IANA - Reserved
101/8   Sep 81   IANA - Reserved
102/8   Sep 81   IANA - Reserved
103/8   Sep 81   IANA - Reserved
104/8   Sep 81   IANA - Reserved
105/8   Sep 81   IANA - Reserved
106/8   Sep 81   IANA - Reserved
107/8   Sep 81   IANA - Reserved
108/8   Sep 81   IANA - Reserved
109/8   Sep 81   IANA - Reserved
110/8   Sep 81   IANA - Reserved
111/8   Sep 81   IANA - Reserved
112/8   Sep 81   IANA - Reserved
113/8   Sep 81   IANA - Reserved
114/8   Sep 81   IANA - Reserved
115/8   Sep 81   IANA - Reserved
116/8   Sep 81   IANA - Reserved
117/8   Sep 81   IANA - Reserved
118/8   Sep 81   IANA - Reserved
119/8   Sep 81   IANA - Reserved
120/8   Sep 81   IANA - Reserved
121/8   Sep 81   IANA - Reserved
122/8   Sep 81   IANA - Reserved
123/8   Sep 81   IANA - Reserved
124/8   Sep 81   IANA - Reserved
125/8   Sep 81   IANA - Reserved
126/8   Sep 81   IANA - Reserved
127/8   Sep 81   IANA - Reserved               See [RFC3330]
128/8   May 93   Various Registries
129/8   May 93   Various Registries
130/8   May 93   Various Registries
131/8   May 93   Various Registries
132/8   May 93   Various Registries
133/8   May 93   Various Registries
134/8   May 93   Various Registries
135/8   May 93   Various Registries
136/8   May 93   Various Registries
137/8   May 93   Various Registries
138/8   May 93   Various Registries
139/8   May 93   Various Registries
140/8   May 93   Various Registries
141/8   May 93   Various Registries
142/8   May 93   Various Registries
143/8   May 93   Various Registries
144/8   May 93   Various Registries
145/8   May 93   Various Registries
146/8   May 93   Various Registries
147/8   May 93   Various Registries
148/8   May 93   Various Registries
149/8   May 93   Various Registries
150/8   May 93   Various Registries
151/8   May 93   Various Registries
152/8   May 93   Various Registries
153/8   May 93   Various Registries
154/8   May 93   Various Registries
155/8   May 93   Various Registries
156/8   May 93   Various Registries
157/8   May 93   Various Registries
```

Statische Konfiguration



Dynamische Konfiguration



```
158/8   May 93   Various Registries
159/8   May 93   Various Registries
160/8   May 93   Various Registries
161/8   May 93   Various Registries
162/8   May 93   Various Registries
163/8   May 93   Various Registries
164/8   May 93   Various Registries
165/8   May 93   Various Registries
166/8   May 93   Various Registries
167/8   May 93   Various Registries
168/8   May 93   Various Registries
169/8   May 93   Various Registries
170/8   May 93   Various Registries
171/8   May 93   Various Registries
172/8   May 93   Various Registries
173/8   Apr 03   IANA - Reserved
174/8   Apr 03   IANA - Reserved
175/8   Apr 03   IANA - Reserved
176/8   Apr 03   IANA - Reserved
177/8   Apr 03   IANA - Reserved
178/8   Apr 03   IANA - Reserved
179/8   Apr 03   IANA - Reserved
180/8   Apr 03   IANA - Reserved
181/8   Apr 03   IANA - Reserved
182/8   Apr 03   IANA - Reserved
183/8   Apr 03   IANA - Reserved
184/8   Apr 03   IANA - Reserved
185/8   Apr 03   IANA - Reserved
186/8   Apr 03   IANA - Reserved
187/8   Apr 03   IANA - Reserved
188/8   May 93   Various Registries
189/8   Apr 03   IANA - Reserved
190/8   Apr 03   IANA - Reserved
191/8   May 93   Various Registries
192/8   May 93   Various Registries
193/8   May 93   RIPE NCC                    (whois.ripe.net)
194/8   May 93   RIPE NCC                    (whois.ripe.net)
195/8   May 93   RIPE NCC                    (whois.ripe.net)
196/8   May 93   Various Registries
197/8   May 93   IANA - Reserved
198/8   May 93   Various Registries
199/8   May 93   ARIN                        (whois.arin.net)
200/8   Nov 02   LACNIC                      (whois.lacnic.net)
201/8   Apr 03   LACNIC                      (whois.lacnic.net)
202/8   May 93   APNIC                       (whois.apnic.net)
203/8   May 93   APNIC                       (whois.apnic.net)
204/8   Mar 94   ARIN                        (whois.arin.net)
205/8   Mar 94   ARIN                        (whois.arin.net)
206/8   Apr 95   ARIN                        (whois.arin.net)
207/8   Nov 95   ARIN                        (whois.arin.net)
208/8   Apr 96   ARIN                        (whois.arin.net)
209/8   Jun 96   ARIN                        (whois.arin.net)
210/8   Jun 96   APNIC                       (whois.apnic.net)
211/8   Jun 96   APNIC                       (whois.apnic.net)
212/8   Oct 97   RIPE NCC                    (whois.ripe.net)
213/8   Mar 99   RIPE NCC                    (whois.ripe.net)
214/8   Mar 98   US-DOD
215/8   Mar 98   US-DOD
216/8   Apr 98   ARIN                        (whois.arin.net)
217/8   Jun 00   RIPE NCC                    (whois.ripe.net)
218/8   Dec 00   APNIC                       (whois.apnic.net)
219/8   Sep 01   APNIC                       (whois.apnic.net)
220/8   Dec 01   APNIC                       (whois.apnic.net)
221/8   Jul 02   APNIC                       (whois.apnic.net)
222/8   Feb 03   APNIC                       (whois.apnic.net)
223/8   Apr 03   IANA - Reserved
224/8-239/8   Sep 81   IANA - Multicast
240/8-255/8   Sep 81   IANA - Reserved
```

● **Private IP-Adressbereiche (Private IPs)**: Für die Verwendung innerhalb von LANs wurden eigene Adressbereiche festgelegt, die nicht geroutet werden. Diese IP-Adressen sind daher auch nicht weltweit eindeutig, sondern nur im jeweiligen lokalen Netzwerk.

Laut RFC 1918 sind für **„private" Netze folgende IP-Bereiche** gestattet (Rechner mit diesen IP-Adressen dürfen keinen direkten Internet-Verkehr haben, d.h. mit dem Internet nur über Proxy-Server in Kontakt treten; sie werden nicht geroutet!):

```
10.0.0.0     – 10.255.255.255  (Class A-Bereich)
172.16.0.0   – 172.31.255.255  (Class B-Bereich)
192.168.0.0 – 192.168.255.255 (Class C-Bereich)
```

5.3 Aufbau von IP-Adressen

Beispiel

```
Adresse              192.168.100.1
Subnetzmaske         255.255.255.0
```

Um TCP/IP Adressen verstehen zu können, muss man sich vor Augen halten, dass die „reale" Schreibweise von Adressen in binärer Form erfolgt (4 Oktetts à 8 Bit).

192	168	100	1
11000000	10101000	01100100	00000001

Gerechnet wird dann wie folgt:

	128	64	32	16	8	4	2	1
192	1	1	0	0	0	0	0	0
168	1	0	1	0	1	0	0	0
100	0	1	1	0	0	1	0	0
1	0	0	0	0	0	0	0	1
	x	x	x	x	x	x	x	x
255	1	1	1	1	1	1	1	1

```
192 = 1100 0000 = 128 + 64
168 = 1010 1000 = 128 + 32  +  8
100 = 0110 0100 = 64  + 32  +  4
1   = 0000 0001 = 1
```

Man hat also mit einer solchen 32-Bit-Adresse insgesamt 232 = 4 294 967 296 Möglichkeiten (also mehr als 4 Milliarden), einen PC unverwechselbar zu adressieren.

IP-Adressen bestehen aus **zwei Teilen**:

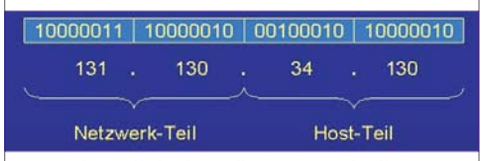
Der erste Teil ist die **Netzwerk-Adresse (Net-ID)**. Da das Internet aus vielen miteinander verbundenen lokalen Netzen (LAN) besteht, ist es sinnvoll, jedem LAN eine eindeutige Adresse zuzuweisen.

Der zweite Teil gibt die Adresse der einzelnen Rechner im Netz an **(Host-Adresse, Host-ID, Knotenadresse)**. Dieser Teil wird durch das lokale Netzwerkmanagement frei vergeben.

Wie viele Bit zur NetID bzw. zur HostID gehören, wird durch die Subnetz-Maske festgelegt. Dafür gibt es folgende einfache Regel:

Ist ein Bit der Subnetzmaske **1**, so gehört das entsprechende Bit der IP-Adresse zur **Net-ID**.

Ist ein Bit der Subnetzmaske **0**, so gehört das entsprechende Bit der IP-Adresse zur **Host-ID**.



Im obigen Beispiel würde also die Subnetzmaske **255.255.0.0** lauten.

Grundsätzlich ist die Länge der Net-ID und der Host-ID frei wählbar. Es haben sich aber zwei verschiedene Sichtweisen bzw. Technologien durchgesetzt:

● klassenorientiertes IP-Routing (fixe Länge von Net-ID und Host-ID)

● „classless IPs" (frei wählbare Länge von Net-ID und Host-ID)

5.4 Klassenorientierte IP-Adressen

Diese Methode basiert auf fix festgelegten Längen für den Net- und den Host-Anteil der IP-Adressen.

Class-A-Netze: Adresse beginnt mit einer binären 0, 7 Bit für Netzwerk-Adresse, 24 Bit für Host-Adresse. Damit gibt es weltweit 127 derartige Netzwerke, ein Class-A-Netz kann bis zu 16 Mio. Teilnehmer haben. Alle derartigen Netzadressen sind bereits belegt.

IP-Adressen von Class-A-Netzen

```
0.0.0.0  bis  127.255.255.255
```

Class-B-Netze: Adresse beginnt mit der binären Ziffernkombination 10, 14 Bit für Netzwerk-Adresse, 16 Bit für Host-Adresse. Damit gibt es weltweit 16384 derartige Netzwerke, ein Class-B-Netz kann bis zu 65536 Teilnehmer haben. Alle derartigen Netzadressen sind bereits belegt.

IP-Adressen von Class-B-Netzen

```
128.0.0.0  bis  191.255.255.255
```

Class-C-Netze: Adresse beginnt mit der binären Ziffernkombination 110, 21 Bit für Netzwerk-Adresse, 8 Bit für Host-Adresse. Damit gibt es weltweit 2 Mio. derartige Netzwerke, ein Class-C-Netz kann bis zu 256 Teilnehmer haben. Neu zugeteilte Netzadressen sind heute immer vom Typ C. Es ist abzusehen, dass bereits in Kürze alle derartigen Adressen vergeben sein werden.

IP-Adressen von Class-C-Netzen

```
192.0.0.0  bis  223.255.255.255
```

Class D-Netze haben einen speziellen Anwendungsbereich (Multicast-Anwendungen) und haben für Internet keine Bedeutung.

Zusammenfassung

CLASS	Netzwerk Anteil	Anzahl Netze	Hostanteil	Anzahl Hosts/Netz
A	1+7 Bit	128	24 Bit	16.777.214
B	2+14 Bit	16.864	16 Bit	65.534
C	3+21 Bit	2.097.152	8 Bit	253

5.5 Besondere IP-Adressen

a) Netzwerkmasken

Netzwerkmasken unterscheiden sich in der Länge des Netzwerk- (alle Bitstellen auf 1) und Hostanteils (alle Bitstellen auf 0)

abhängig von der Netzwerkklasse

	1.Byte	2.Byte	3.Byte	4.Byte
Class A	255	0	0	0
Class B	255	255	0	0
Class C	255	255	255	0

Netzwerkmasken stellen einen Filter dar, an dem Rechner entscheiden können, ob sie sich im selben (logischen) Netz befinden

b) Netzwerkadressen

Die Netzwerkadresse eines Rechners ergibt sich, indem man die IP-Adresse mit der Netzwerkmaske bitweise UND-verknüpft. Generell gilt, dass bei Netzwerkadressen alle Bitstellen des Hostanteils 0 sind.

Host-adresse	192.168.100.1	11000000	10101000	01100100	00000001
UND					
Maske	255.255.255.0	11111111	11111111	11111111	00000000
Subnetz	192.168.100.0	11000000	10101000	01100100	00000000

Nur Rechner mit der gleichen Netzwerkadresse befinden sich im gleichen logischen Netzwerk!

c) Broadcast-Adresse

Die Broadcast-Adresse ergibt sich aus der IP-Adresse, bei der alle Bitstellen des Hostanteils auf 1 gesetzt sind. Sie bietet die Möglichkeit, Datenpakete an alle Rechner eines logischen Netzwerkes zu senden. Sie wird ermittelt, indem die Netzwerkadresse mit der invertierten Netzwerkmaske bitweise ODER-verknüpft wird.

Beispiel

Subnetz	192.168.100.0	11000000	10101000	01100100	00000000
ODER					
invertierte Maske	0.0.0.255	00000000	00000000	00000000	11111111
Broadcast	192.168.100.255	11000000	10101000	01100100	11111111

d) Loopback-Adresse

Die Class-A-Netzwerkadresse 127 ist weltweit reserviert für das sogenannte *local loopback*; sie

dient zu Testzwecken der Netzwerkschnittstelle des eigenen Rechners.

Die IP-Adresse `127.0.0.1` ist standardmäßig dem Loopback-Interface jedes Rechners zugeordnet. Alle an diese Adresse geschickten Datenpakete werden nicht nach außen ins Netzwerk gesendet, sondern an der Netzwerkschnittstelle reflektiert.

Die Datenpakete erscheinen, als kämen sie aus einem angeschlossenem Netzwerk.

5.6 Subnetting

Internet-Quellen:

- *http://instrumentation.de/5106003d.htm*
- *http://www.zyxel.de/support*

Das obige Schema zeigt, dass nur eine begrenzte Anzahl an internationalen IP-Adressen verfügbar ist. Falls die Anzahl der Netzwerke nicht ausreicht, gibt es wie schon erwähnt, die Möglichkeit diese Anzahl durch geschickte Strukturierung von Subnetzen zu erweitern. In der folgenden Tabelle ist eine mögliche Unterteilung dargestellt.

Subnetzmaske	Anzahl Subnetze (*)	Anzahl Hosts (Rechner, Knoten)
`255.255.255.0`	1 (1)	254
`255.255.255.128`	0 (2)	126
`255.255.255.192`	2 (4)	62
`255.255.255.224`	6 (8)	30
`255.255.255.240`	14 (16)	14
`255.255.255.248`	30 (32)	6
`255.255.255.252`	62 (64)	2

(*) Die in Klammer stehenden Werte sind zwar rechnerisch möglich, enthalten aber u.U. verbotene Adressen (s.u.).

Wie daraus die möglichen Netze und zugehörigen gültigen IP-Adressen entstehen, soll am Beispiel der Subnetzmasken `255.255.255.192` und `255.255.255.224` erläutert werden. Der Status erlaubt oder nicht ergibt sich daraus, dass die erste und letzte bei der Unterteilung entstehenden Adressen nicht verwendet werden dürfen.

Netze und IP-Adressen mit Subnetz-Maske `255.255.255.192`:

Netzwerkadresse	IP-Adressen	Broadcast	Status
`a.b.c.0`	1-62	63	nicht erlaubt (*)
`a.b.c.64`	65-126	127	erlaubt
`a.b.c.128`	129-190	191	erlaubt
`a.b.c.192`	193-254	255	nicht erlaubt (*)

(*) Anmerkung: Es ist nicht sofort einsichtig, warum das erste und das letzte Subnet „nicht erlaubt" sind. Der Grund dafür liegt in der Tatsache, dass im vorliegenden Beispiel ein Class C-Netz unterteilt wurde. Class C-Netze haben ohne Subnetting eine Subnetz-Maske `255.255.255.0`, wobei sich aus den vorher erwähnten Regeln ergibt, dass die IP-Adresse `a.b.c.0` (also alle Bit der HostID auf 0 gesetzt) der Netzwerkadresse entspricht und diese (einzige) Adresse daher nicht verwendet werden darf. Bei der Unterteilung in Subnetze zeigt sich aber, dass beim gesamten Bereich von `a.b.c.0` bis `a.b.c.63` die SubnetID aus lauter Nullen besteht – daher der ganze Bereich ausfällt. Die Argumentation für das letzte Subnetz ist analog zu sehen.

Netze und IP-Adressen mit Subnetz-Maske `255.255.255.224`:

Netzwerkadresse	IP-Adressen	Broadcast	Status
`a.b.c.0`	1-30	31	nicht erlaubt
`a.b.c.32`	33-62	63	erlaubt
`a.b.c.64`	65-94	95	erlaubt
`a.b.c.96`	97-126	127	erlaubt
`a.b.c.128`	129-158	159	erlaubt
`a.b.c.160`	161-190	191	erlaubt
`a.b.c.192`	193-222	223	erlaubt
`a.b.c.224`	225-254	255	nicht erlaubt

Spätestens bei der Einrichtung eines Netzwerkes mit Subnetzen dürfte klar werden, dass hier eine ganze Menge Fehlerquellen schlummern und dass gute Netzwerkadministratoren durchaus Ihre Daseinsberechtigung haben! Man sollte deshalb bei Problemen neuer Rechner/Geräte im Netzwerk die Adressen sehr genau überprüfen.

5.7 CIDR (Classless Inter-Domain Routing), VLSM (Variable Length Subnet Masks) und Supernetting

Das CIDR beschreibt ein Verfahren zur effektiveren Nutzung der bestehenden 32 Bit umfassenden IP-Adresse. Bei diesem Verfahren werden IP-Adressen zusammengefasst, wobei ein Block von aufeinander folgenden IP-Adressen der Klasse C als ein Netzwerk behandelt werden.

Möglich wird dies durch "Kürzen" der NetID, die bei klassenorientierter Betrachtung 24 Bit lang wäre. Man verwendet daher Netzwerke wie etwa `192.168.4.0/23` mit insgesamt 510 gültigen Host-Adressen.

Das CIDR-Verfahren reduziert die in Routern gespeicherten Routing-Tabellen durch einen Präfix in der IP-Adresse. Mit diesem Präfix kann ein großer Internet Service Provider bzw. ein Betreiber eines großen Teils des Internets gekennzeichnet werden. Dadurch können auch darunter liegende Netze zusammengefasst werden; so genanntes Supernetting. Die Methode wird in **RFC 1518** beschrieben.

Um einen Mangel an Netzwerkkennungen zu verhindern, haben Internetinstitutionen ein Schema erarbeitet, das so genannte Supernetting. Im Gegensatz zum Subnetting werden beim Supernetting Bits der Netzwerkkennung verwendet und für effizienteres Routing als Hostkennung maskiert. Statt einer Organisation mit 2.000 Hosts eine Netzwerkkennung der Klasse B zuzuweisen, weist ARIN *(American Registry for Internet Numbers)* beispielsweise einen Bereich von acht Netzwerkkennungen der Klasse C zu. In jeder Netzwerkkennung der Klasse C sind 254 Hosts möglich. Dies ergibt insgesamt 2.032 Hostkennungen.

Beispiel

Ohne Supernetting

Routingtabelle für Router B

`220.78.168.0`	`255.255.255.0`	`220.78.168.1`
`220.78.169.0`	`255.255.255.0`	`220.78.168.1`
`220.78.170.0`	`255.255.255.0`	`220.78.168.1`
`220.78.171.0`	`255.255.255.0`	`220.78.168.1`
`220.78.172.0`	`255.255.255.0`	`220.78.168.1`
`220.78.173.0`	`255.255.255.0`	`220.78.168.1`
`220.78.174.0`	`255.255.255.0`	`220.78.168.1`
`220.78.175.0`	`255.255.255.0`	`220.78.168.1`

Mit Supernetting

Routingtabelle für Router B

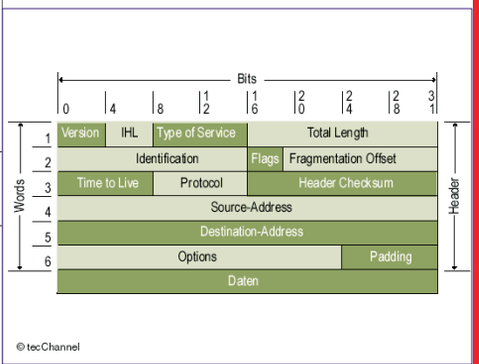
`220.78.168.0`	`255.255.252.0`	`220.78.168.1`

5.8 Aufbau des IP-Headers

Im Internet gibt es die Seite `www.protocols.com`, auf der detailliert eine ganze Reihe von Netzwerkprotokollen beschrieben sind – darunter auch das TCP/IP-Protokoll.

Wir haben bereits erwähnt, dass jedes Protokoll spezielle Informationen (den so genannten Header) zu den eigentlichen Daten hinzufügt.

Wir wollen hier den IP-Header etwas genauer betrachten. Zuerst sollen an dieser Stelle das Aussehen und die Bedeutung der einzelnen Header-Elemente beschrieben werden.



Die ersten vier Bits stellen das Feld **Ver** dar (**siehe Abbildung**). Sie sind für die Version des IP-Protokolls bestimmt, welches das zu sendende Datagramm zusammenstellt. Bei der Benutzung von IPv4 enthält dieses Feld den Wert vier.

Die nächsten vier Bit, die das Feld **HLen** repräsentieren, enthalten die aktuelle Header-Länge. Dabei werden aber nicht die Bytes, sondern die Doppel-Worte (4 Byte) gezählt. Bei einem IP-Standard-Header sollte hier eine fünf stehen. Dieser Standard-Header findet bei der Übertragung normaler Nutzdaten Anwendung. Er umfasst immer 5 Doppel-Worte = 20 Byte.

Danach folgt das Feld **TOS**, *Type of Service*. Es enthält u.a. Informationen, welcher Art die zu transportierenden Daten sind und welche Qualität die Art der Übertragung besitzen soll.

Das Feld **Total Length** im IP-Header kennzeichnet die totale Länge eines Datagramms einschließlich Header. Da dieses Feld nur eine 16-Bit-Zahl enthalten kann, ist auch die Größe eines IP-Datagramms auf maximal $2^{16} - 1 = 65535$ Byte beschränkt. Ein größeres Datagramm kann durch IP nicht vermittelt werden.

Im Zuge der QoS (Quality of Service)-Diskussion (Ziel: Qualitätsverbesserung der Internet-Protokolle und Internet-Dienste) am Internet wurde eine Lösung ersonnen, die als **„diffserv"** *(differentiated services)* bezeichnet wird. Diffserv (DS) baut am TOS-Feld auf und überträgt in diesem Byte Informationen, die das Routing effizienter machen.

Auf die Bedeutung der Felder **Identification, Flags** und **Fragment Offset** wird später näher eingegangen. Sie werden benötigt, um eine Datagramm-Übermittlung auch über Netzverbindungen zu garantieren, die die maximale Größe eines IP-Datagramms nicht transportieren können.

Im Feld **TTL** wird die Lebenszeit, *Time To Live*, eines Datagramms verwaltet. Es dient zur Vorbeugung, dass ein Datagramm im Netz nicht "ewig herumirrt". Beim Verschicken des Datagramms wird durch den Sender eine Zahl in dieses Feld eingesetzt, die die Lebenszeit die-

ses Datagramms in Sekunden repräsentieren soll. Da aber ein anderer Host nicht weiß, wann dieses Datagramm erzeugt wurde und im Header auch keine Information über die Erzeugung vorhanden ist, repräsentiert diese Zahl in der Praxis etwas anderes. Sie gibt an, wie viele Router dieses Datagramm passieren darf, um den Empfänger zu erreichen. Dazu ist es notwendig, dass jeder benutzte Router den Wert dieses Feld um 1 erniedrigt. Ist irgendwann einmal der Wert des Feldes **TTL** gleich Null, dann wird es von dem Router, der es gerade bearbeitet, verworfen, und er sendet eine Fehlermeldung zurück an den Sender.

Das Feld **Protocol** wird von IP benutzt, um auf der Seite des Senders das Protokoll zu vermerken, welches die Dienste von IP in Anspruch nimmt. Auf der Seite des Empfängers dient es IP dazu, das Datagramm genau an dieses Protokoll zur weiteren Bearbeitung weiterzuleiten.

Das Feld **Header Checksum** beinhaltet eine Prüfsumme. Sie dient zum Erkennen von Verfälschungen bei der Übertragung des Datagramms. Allerdings wird sie nur über die Daten des IP-Headers selbst gebildet. Die zu transportierenden Daten werden nicht berücksichtigt. Soll über diesen Daten auch eine Prüfsumme zur Fehlererkennung gebildet werden, muss das ein anderes Protokoll oder die Anwendung selbst übernehmen, die die Dienste von IP in Anspruch nimmt. Die Überprüfung ist einfach zu vollziehen. Der das Datagramm bearbeitende Host, das kann auch ein Router sein, extrahiert den Wert aus dem Feld *Header Checksum* des Datagramms und berechnet diesen neu. Gleichen sich die beiden Werte nicht, wird IP dieses Datagramm verwerfen und eine Fehlermeldung an den Sender schicken. Ansonsten wird das Datagramm an den Empfänger zugestellt. Der Algorithmus zur Erstellung dieser Prüfsumme ist recht simpel. Der Wert dieser Prüfsumme stellt das Einerkomplement der Einerkomplementsumme des Headers dar. Dabei werden die Daten in Einheiten von 16 Bit zerteilt und addiert. Zur Berechnung wird der Header vollständig ausgefüllt. Das Feld *Header Checksum* wird vor der Berechnung mit Null initialisiert. Als Eingabe des Algorithmus bei einem Standard-Header dienen dann diese so vorbereiteten 20 Byte = 10 Worte. Das ermittelte Ergebnis wird zuletzt in das Feld **Header Checksum** übertragen. Der Grund, nur über den IP-Header eine Prüfsumme zu bilden, liegt darin begründet, dass diese Berechnung auf jedem Router durchgeführt werden muss. Dieses Verfahren stellt gegenüber der Berechnung über alle Daten eine erhebliche Beschleunigung der Vermittlung dar.

Zur Adressierung des Datagramms werden unbedingt die zwei Felder **Source IP Address** (Quell-Adresse) und **Destination IP Address** (Ziel-Adresse) benötigt. Die Ziel-Adresse dient zur Adressierung des Empfängers. Das Eintragen einer Quell-Adresse wird einmal zur etwaigen Erzeugung von Fehlermeldungen benötigt und außerdem dient sie dem Empfänger zur Identifizierung des Senders.

Im Feld **Data** können alle möglichen Nutzdaten transportiert werden.

Die Felder **IP Options** und **Padding** hängen direkt miteinander zusammen. Da der IP-Header immer Vielfache von Doppel-Worten enthalten muss, die Optionen aber verschieden lang sein können, wird das **Padding** zur Auffüllung genutzt, um wieder ein volles Doppel-Wort zu erhalten. Wird durch IP festgestellt, dass der Wert im Feld **HLen** größer als 5 ist, muss der Header Optionen enthalten. An Hand dieser Header-Länge ist auch ersichtlich, wo die Optionen enden und von wo ab eventuell Daten im Datagramm enthalten sind. Die Bedeutung der Optionen werden u.a. im **RFC 791** beschrieben.

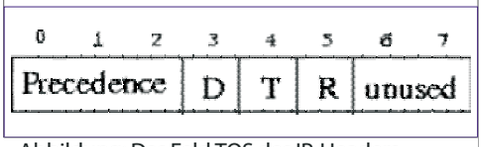


Abbildung: Das Feld TOS des IP-Headers

Die Abbildung zeigt den Aufbau des Feldes **TOS**. Die drei Bits des Feldes **Precedence** kennzeichnen die Art des Datagramms. Sie können einen Wert zwischen 0 und 7 annehmen. Der Wert 0 wird bei einem Datagramm eingesetzt, welches normale Nutzdaten transportiert. Der Wert 7 wird für Datagramme zur Netzwerk-Steuerung verwendet. Näheres dazu ist im **RFC 791** zu erfahren. Die Felder **D**, **T** und **R** legen fest, welcher Qualität die Art der Übertragung des Datagramms sein soll. Feld **D** macht dabei eine Aussage über die Schnelligkeit, Feld **T** über den Durchsatz und Feld **R** über die Verfügbarkeit der Übertragung. Setzt z.B. ein Sender das Bit in Feld **D** in einem Datagramm, verlangt er, dass dieses so schnell wie möglich an den Empfänger übermittelt wird.

Der Header muss grundsätzlich in der Netzwerk-Byte-Ordnung *(network byte order)* verschickt werden. Diese Ordnung wird auch *Big Endian* genannt.

5.9 IP-Rechner

Auf den folgenden Seiten finden Sie IP-Adressrechner zum Download, aber auch Rechner, die Sie online einsetzen können:

http://www.chinet.com/html/ip.html

http://www.tmp-houston.com/subcalc.htm

http://jodies.de/ipcalc

http://www.telusplanet.net/public/sparkman/netcalc.htm

http://www.wildpackets.com/products/ipsubnetcalculator

http://www.novell.com/coolsolutions/tools/1466.html

5.10 IPv6

Quelle: *http://www.ipv6-net.de*

Man arbeitet bereits seit längerer Zeit an einem neuen Standard (Version 6 des Internet Protokolls, **IPv6** oder **IPng** für *„next generation“*), der statt einer Adresslänge von 32 bit eine Länge von 128 bit haben soll. Um die Kompatibilität zu gewährleisten, wird die IPv4-Adresse in der neuen Adresse "enthalten sein".

Windows Server 2003 unterstützt bereits IPv6.

IPv6 verwendet zur Darstellung seiner IP-Adressen das Hexadezimalsystem in einer Adresslänge von 128 Bit. Eine solche IPv6-Adresse könnte beispielsweise so aussehen:

```
3ffe:400:89AB:381C:7716:AA91:0000:0001
```

Um eine IPv6-Adresse wie die angegebene verkürzt darzustellen, kann man auf die Nullen in einer Gruppe verzichten:

```
3ffe:400:89AB:381C:7716:AA91::1.
```

Ein weiterer Vorteil von IPv6 ist die, gegenüber IPv4 stark vereinfachte Headerstruktur, die eine merkbar schnellere Bearbeitung am den Router ermöglicht.

Aufbau des IPv6-Header

```
<----------------- Bits ---------------------->
                    1   1   2   2   2   3
0   |    4    | 8 | 2 | 6 | 0 | 4 | 8 | 1
-------------------------------------------
1 |   Version |  Class |         Flow-Label     |
-------------------------------------------
2 |        Payload Lenght      |  Next |Hop-Limit|
-------------------------------------------
3 |                Source Address               |
4 |                                             |
-------------------------------------------
5 |             Destination Address             |
6 |                                             |
-------------------------------------------
```

Version (4 Bit)

Enthält immer den Wert '6' bei IPv6. Dieses Feld dient der Software zur Unterscheidung verschiedener IP-Versionen.

Class (8 Bit)

Gibt die Priorität der zu übermittelnden Daten an.

Flow-Label (20 Bit)

Dieses Feld kennzeichnet einen Datenstrom zwischen Sender und Empfänger. Alle Pakete die zu einem bestimmten Datenstrom gehören, tragen in diesem Feld den gleichen Wert.

Payload Length (16 Bit)

Hier wird die Länge des Datenpakets (nach dem ersten Header) angegeben.

Next (8 Bit)

Gibt den Typ des nächsten Headers an. Der Wert '59' signalisiert, dass keine weiteren Header bzw. Daten folgen.

Hop-Limit (8 Bit)

Legt fest, nach wie vielen Durchgängen das Paket vom Router, zur Vermeidung von Schleifen, verworfen werden soll.

Source Address (128 Bit)

Beinhaltet die Absenderadresse.

Destination Address (128 Bit)

Beinhaltet die Empfängeradresse.

Im Moment unterstützen besonders europäische und asiatische Institutionen und Firmen die Entwicklung und Verbreitung von IPv6. Das ist wohl mit der Tatsache, dass etwa 75% des IPv4-Adressraums den USA zugeteilt wurde, zu erklären. Im Moment unterstützen zwar nur wenige Dienste das Internet Protokoll der Zukunft, aber gerade bei der Entwicklung neuer Dienste in diesem Bereich wird es in den nächsten Jahren einen enormen Zuwachs geben.

5.11 ARP (Address Resolution Protocol)

Das *Address Resolution Protocol* (ARP) arbeitet auf der Schicht 2, der Sicherungsschicht, des OSI-Schichtenmodells und setzt IP-Adressen in Hardware- und MAC-Adressen um. Alle Netzwerktypen und -topologien benutzen Hardware-Adressen um die Datenpakete zu adressieren. Damit nun ein IP-Paket an sein Ziel findet, muss die Hardware-Adresse des Ziels bekannt sein.

Jede Netzwerkkarte besitzt eine einzigartige und eindeutige Hardware-Adresse, die fest auf der Karte eingebrannt ist und meist nicht änderbar ist, die *Media Access Control*-Adresse oder kurz **MAC-Adresse**. In Ethernet-Netzwerken ist diese Adresse meist eine 48 Bit-Binärzahl,

die als 6 hexadezimal angegebenen Bytes angeschrieben wird.

Bevor nun ein Datenpaket verschickt werden kann, muss durch ARP eine Adressauflösung erfolgen. Dazu benötigt ARP Zugriff auf IP-Adresse und Hardware-Adresse. Um an die Hardware-Adresse einer anderen Station zu kommen verschickt ARP z. B. einen Ethernet-Frame als Broadcast-Meldung mit der MAC-Adresse `"FF FF FF FF FF FF"`. Diese Meldung wird von jedem Netzwerkinterface entgegengenommen und ausgewertet. Der Ethernet-Frame enthält die IP-Adresse der gesuchten Station. Fühlt sich eine Station mit dieser IP-Adresse angesprochen, schickt sie eine ARP-Antwort an den Sender zurück. Die gemeldete MAC-Adresse wird dann im lokalen ARP-Cache des Senders gespeichert. Dieser Cache dient zur schnelleren ARP-Adressauflösung.

Ablauf einer ARP-Adressauflösung

Eine ARP-Auflösung unterscheidet zwischen lokalen IP-Adressen und IP-Adressen in einem anderen Subnetz. Als erstes wird anhand der Subnetzmaske festgestellt, ob sich die IP-Adresse im gleichen Subnetz befindet. Ist das der Fall, wird im ARP-Cache geprüft, ob bereits eine MAC-Adresse für die IP-Adresse hinterlegt ist. Wenn ja, dann wird die MAC-Adresse zur Adressierung verwendet. Wenn nicht, setzt ARP eine Anfrage mit der IP-Adresse nach der Hardware-Adresse in das Netzwerk. Diese Anfrage wird von allen Stationen im selben Subnetz entgegengenommen und ausgewertet. Die Stationen vergleichen die gesendete IP-Adresse mit ihrer eigenen. Wenn sie nicht übereinstimmt, wird die Anfrage verworfen. Wenn die IP-Adresse übereinstimmt schickt die betreffende Station eine ARP-Antwort direkt an den Sender der ARP-Anfrage. Dieser Speichert die Hardware-Adresse in seinem Cache. Da bei beiden Stationen die Hardware-Adresse bekannt sind, können sie nun miteinander Daten austauschen.

Befindet sich eine IP-Adresse nicht im gleichen Subnetz, geht ARP über das Standard-Gateway. Findet ARP die Hardware-Adresse des Standard-Gateways im Cache nicht, wird eine lokale ARP-Adressauflösung ausgelöst. Ist die Hardware-Adresse des Standard-Gateways bekannt, schickt der Sender bereits sein erstes Datenpaket an die Ziel-Station. Der Router (Standard-Gateway) nimmt das Datenpaket in Empfang und untersucht den IP-Header. Der Router überprüft, ob sich die Ziel-IP-Adresse in einem angeschlossenen Subnetz befindet. Wenn ja, ermittelt er anhand der lokalen ARP-Adressauflösung die MAC-Adresse der Ziel-Station. Anschließend leitet er das Datenpaket weiter. Ist das Ziel in einem entfernten Subnetz, überprüft der Router seine Routing-Tabelle, ob ein Weg zum Ziel bekannt ist. Ist das nicht der Fall steht dem Router auch ein Standard-Gateway zu Verfügung. Der Router führt für sein Standard-Gateway eine ARP-Adressauflösung durch und leitet das Datenpaket an dieses weiter.

Die vorangegangenen Schritte weiderholen sich dann so oft, bis das Datenpaket sein Zeil erreicht oder das IP-Header-Feld TTL auf den Wert 0 springt. Dann wird das Datenpaket vom Netz genommen.

Erreicht dann irgendwann das Datenpaket doch sein Ziel, schreibt die betreffende Station seine Rückantwort in ein ICMP-Paket an den Sender. In dieser Antwort wird falls möglich ein Gateway vermerkt, über das die beiden Stationen miteinander kommunizieren. So werden weitere ARP-Adressauflösungen und dadurch Broadcasts vermieden.

ARP-Cache

Anzeigen des ARP-Caches unter Windows 2000/XP/2003:

```
C:\>arp -a
Schnittstelle: 192.168.168.11 --- 0x2
Internetadresse    Physikal. Adresse  Typ
192.168.168.8      00-30-ab-0e-d3-6a  dynamisch
```

Durch den ARP-Cache wird vermieden, dass bei jedem Datenpaket an das selbe Ziel wieder und immer wieder ein ARP-Broadcast ausgelöst wird. Häufig benutzte Hardware-Adressen sind im ARP-Cache gespeichert. Die Einträge im ARP-Cache können statisch oder dynamisch sein. Statische Einträge können manuell hinzugefügt und gelöscht werden. Dynamische Einträge werden durch die ARP-Adressauflösung erzeugt.

Jeder dynamische Eintrag bekommt einen Zeitstempel. Ist er nach zwei Minuten nicht mehr abgerufen worden, wird der Eintrag gelöscht. Wird eine Adresse auch nach zwei Minuten noch benutzt, wird der Eintrag erst nach zehn Minuten gelöscht. Ist der ARP-Cache für neue Einträge zu klein, werden alte Einträge entfernt.

Wird die Hardware neu gestartet oder ausgeschaltet, wird der ARP-Cache gelöscht. Es gehen dabei auch die statischen Einträge verloren.

Fehler und Probleme mit ARP: Grundsätzlich gibt es keine Probleme oder Fehler mit ARP, solange keine statischen Einträge im ARP-Cache vorgenommen werden oder Hardware-Adressen von Netzwerkkarten verändert werden.

ARP läuft für den Benutzer ganz im Verborgenen.

Den umgekehrten Weg, MAC-Adresse bekannt, IP-Adresse gesucht, definiert RARP *(Reverse Adress Resolution Protocol)*.

Vorsicht...

Aus einer Aussendung des TGM-Netzwerkteams

Ercan Karaduman, Berndt Sevcik

...am lokalen Computer

- Computer nie unbeaufsichtigt lassen
- Bildschirmsperre auch bei kurzer Abwesenheit
- Passwörter komplex aussuchen, nicht weitergeben und regelmäßig ändern
- Passwörter nicht am und in der Nähe des Computers aufbewahren
- Netzwerkfreigaben überprüfen und vergessene Freigaben wieder deaktivieren
- Keine Disketten oder CD's im Laufwerk lassen
- Bei heiklen Daten, Ordnerberechtigungen überprüfen (Deaktivieren von Jeder - Vollzugriff)
- Optional eine Personal-Firewall verwenden (Firewall-Kenntnisse unbedingt erforderlich)
- Virenschutz aktuell halten
- Spyschutz aktuell halten
- Sicherheitspatches für das Betriebssystem aktuell halten
- Systemauslastung bei Verdacht beobachten (Strg+Alt+Entf) Taskmanager - Systemleistung
- Ungewollte Prozesse überprüfen (Strg+Alt+Entf) Taskmanager - Prozesse
- Nicht benötigte Dienste deaktivieren.

...im Netz

- Achtung auf Phishing
- Keine gleichen Passwörter für verschiedene Mailsysteme verwenden (z.B.: gmx und ccc)
- Nicht überall im Internet registrieren. Impressum und Seriosität überprüfen
- Keine persönlichen Daten in Formulare auf unseriösen Webseiten eintragen
- Nicht alle Tools halten was sie versprechen. Achtung: Darunter sehr viele Trojaner
- Cookies löschen
- Verlauf löschen
- Illegale Webseiten vermeiden