



IP-Protokoll

Netze :

IP-Adressen werden im QDN-Format (quad dotted notation) angegeben :

Beispiel : 62.245.200.166

Das erste Byte besagt in der Originalversion des Protokolls, ob es sich bei der Adresse um eine Class A, Class B oder Class C – Adresse handelt. Dies gibt an, welcher Teil der Adresse die Netzadresse („Vorwahl“), und welcher Teil die Rechneradresse in diesem Netz ist.

Class A (erstes Byte zwischen 1 und 127)	: erstes Byte
Class B (erstes Byte zwischen 128 und 191)	: zwei Byte
Class C (erstes Byte zwischen 192 und 223)	: drei Byte



Netmask :

Diese Aussage kann auch durch die netmask angegeben werden, die bei den den Stellen der Netzadresse (netbit) 1, bei denen der Rechneradresse (hostbit) 0 ist :

netmask Class A : 255.0.0.0

netmask Class B : 255.255.0.0

netmask Class C : 255.255.255.0

Die netmask wird auch kürzer geschrieben, indem man einfach die Anzahl der 1-en in der netmask neben der IP-Adresse hinter einem Schrägstrich angibt :

204.66.23.2 / 24 ist z.b. eine Adresse in einem Class-C Netz

Broadcast :

In einem Netz können für bestimmte Funktionen alle Rechner zugleich angesprochen werden, man nennt dies einen Broadcast. Die Broadcastadresse in einem Netz wird gebildet, indem man alle hostbit = 1 setzt.

Netadress :

Das Netz selbst (eigentlich das Kabel..) hat die Adresse, bei der nach der Netzadresse alle hostbit = 0 gesetzt werden. Hier sind nun lokale Netze zu sehen, links das Biernetz, rechts das Weinnetz.



Die Hosts im Biernetz haben die Adressen :

Pils : 182.166.17.4 / 16

bock: 182.166.18.5 / 16

Weizen : 182.166.18.6 / 16

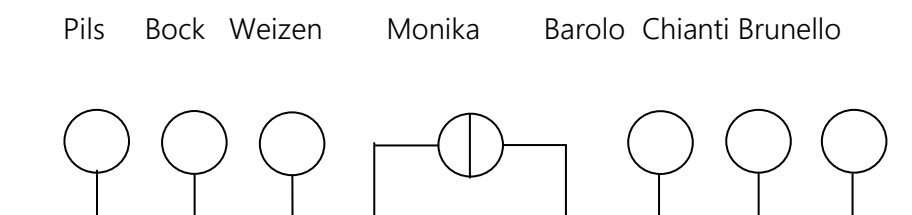
Die Hosts im Weinnetz haben die Adressen :

Barolo : 62.245.200.162 / 8

Chianti : 62.245.123.2 / 8

Brunello : 62.245.1.1 / 8

Der Router Monika zwischen den beiden lokalen Netzen hat links die Adresse 182.166.0.1 und rechts die Adresse 62.0.0.1 (jeweils die ersten IPs in den lokalen Netzen).



Jeder Rechner hat nun eine Tabelle konfiguriert, in der seine IP-Nachbarschaft angegeben ist. Diese Tabelle heißt routing-table.



Routing-table von Pils :

Net	Netmask	Gateway	Interface
182.166.0.0	255.255.0.0	--	eth0
62.0.0.0	255.0.0.0	182.166.0.1	eth0
↑	↑	↑	↑
erreichbares Netz	seine Netmask	der nötige Router	die richtige Netzkarte

Routing-table von Monika :

Net	Netmask	Gateway	Interface
182.166.0.0	255.255.0.0	--	eth0
62.0.0.0	255.0.0.0	--	eth1

(Die Netzkarten werden beim booten ab 1 hochgezählt)



Jetzt will Pils eine Nachricht an Weizen schicken :

➔ ping 182.166.18.6

routing-protokoll

Das IP-Protokoll im Layer 3 von Pils prüft nun anhand der Routing-table, ob Weizen lokal erreichbar ist, oder ob ein Router beauftragt werden muß. Hierzu vergleicht er Zeile für Zeile, indem er die Zieladresse 182.166.18.6 mit der in der Zeile stehenden netmask bitweise parallel UND-verknüpft (ein 32-bit Prozessor kann das in einem einzigen Befehl), und das Ergebnis mit der in der ersten Spalte stehenden netadress vergleicht. Hier ist schon die erste Zeile erfolgreich.



arp-protokoll

Nun muß der Frame aber im Layer 2 als Ziieldresse die MAC von Weizen bekommen, und die ist momentan noch unbekannt. Layer 3 gibt die Ziel-IP 182.166.18.6 an das Arp-Protokoll. Das ARP-Protokoll schickt nun einen Layer 2 – Broadcast ins Netz mit dem Inhalt : Who has 182.166.18.6 ?

Jeder Rechner vergleicht in Layer 3, Weizen sieht, daß er gemeint ist und ruft zurück : It's me !

Damit hat er sich verraten, weil an dem Antwortframe seine MAC dransteht. Das ARP-Protokoll trägt nun den Zusammenhang IP-MAC für Weizen in eine Cache-Tabelle ein (ARP-Cache), so daß beim nächsten Frame nicht wieder gefragt werden muß. Der Ethernetframe kann nun gebildet werden, der Ping geht an Weizen.

Jetzt will Pils eine Nachricht an Brunello schicken : ping 62.245.1.1

Der Ablauf ist der Gleiche wie vorher, nur erkennt Pils in seiner Routing-table, daß er Monika beauftragen muß. Er schickt (nach ARP auf Monikas IP) Monika den Frame. Monika prüft nun in ihrer Routing-table, wohin der Frame muß, und sieht in Zeile 2 das richtige Netz. Sie setzt auf Eth1 (ihre zweite Netzkarte) einen ARP-Request auf Brunello ab, und erhält seine MAC, dann schickt sie ihm das Paket von Pils.



subnetting

IP-Adressen wie oben beschrieben (Class A,B oder C) werden heute kaum noch benutzt. Man teilt diese Netze in kleinere Teilnetze auf, dieser Vorgang wird Subnetting genannt. Hierzu wird die netmask, also die Grenze zwischen Netzteil und Hostteil in der IP, nicht mehr an der Bytegrenze sondern beliebig innerhalb der QDN gezogen. Wenn man die IP nicht in Byteschreibweise, sondern binär angibt, ist das nichts besonderes

62.245.200.163 / 29 ist kompliziert zu handhaben, im letzten Byte verläuft die Grenze.

11111000.11110101.11001000.10100 011 | lautet diese IP binär

Netzadresse (dezimal 62.245.200.160) Rechner

Wieviele Rechner sind in diesem Netz möglich ? Mit 3 Bit sind 8 Adressen möglich, eine davon ist die Netadresse, eine der Broadcast, also sind noch 6 IP für Rechner frei. (Vermutlich wird aber noch eine davon für einen Router benötigt)

Layer 4

Hier werden aus den Datenströmen von oben (z.B. eine Programmdatei oder ein Visualisierungsbild) Teilstücke generiert, die in den Frames im Layer 2 Platz haben. Dazu wird ein System aufgebaut, das die Übertragung absichert (Fehlerkorrektur) und beim Empfänger im Layer 4 die Rekonstruktion des Datenstroms ermöglicht.

Die Layer 5 und 6 haben in heutigen Industrieanwendungen keine große Bedeutung.

Layer 7

In den ersten Feldbussystemen wie MAP oder später Profibus FMS waren in Layer 7 Dienste (Programme) realisiert, die dem Anwendergerät (meist einem einfachen Microcontroller) und damit dem Programmierer beim Anbieter eines Feldgeräts (z.B. ein Mitarbeiter bei einem Ventilhersteller), eine einfache Befehlschnittstelle zum Bussystemen anbietet.

Es muß nicht die hochkomplexe Verwaltung eines Token oder ähnlich programmiert werden, sondern einfache Befehle wie read oder write oder ähnlich.

Fehlt Layer 7, kann ein kleines oder mittleres Unternehmen seine Feldgeräte nur mit Hilfe eines großen Marktteilnehmers feldbusfähig machen.