

Matura

NWES-SESD

Höhere Technische Bundeslehr- und Versuchsanstalt Anichstraße

Abteilung für Wirtschaftsingenieure/Betriebsinformatik

Ausgeführt im Schuljahr 2024 von:
Gwercher

Inhaltsverzeichnis

| | |
|--|-----------|
| I 2BHWII | 1 |
| 1 Digitaltechnik | 2 |
| 1.1 Boolsche Algebra | 3 |
| II 3BHWII | 5 |
| 1.2 Bussysteme | 6 |
| 1.2.1 RS-232 (UART) | 7 |
| 1.2.2 CAN-Bus (Controll Area Network) | 8 |
| 1.2.3 I2C-Bus (IIC, I ² C) | 10 |
| 2 Finite State Machine (Endlicher Automat) | 13 |
| III 4BHWII | 15 |
| 3 IPv4 (Internet Protocol Version 4): | 16 |
| 4 Netzwerke im Alltag und Grundbegriffe | 18 |
| 4.1 Referenzmodell (OSI und TCP/IP) | 20 |
| 4.1.1 Layer 1 (Physical) | 22 |
| 4.1.2 Layer 2 (Data Link) | 24 |
| 4.1.3 Layer 3 (Network) | 28 |
| 4.1.4 Layer 4 (Transport) | 32 |
| 4.1.5 Layer 5, 6, 7 (Session, Presentation, Application) | 36 |
| 4.2 VLANs | 40 |
| IV 5BHWII | 43 |
| 5 Routing | 44 |
| 6 Aufbau des Internets | 47 |
| 7 IPv6 | 48 |
| 8 WLAN (Wireless Local Area Network) | 52 |
| 8.1 Wifi (802.11) | 53 |
| 9 Network Security | 58 |
| 9.1 Firewall | 59 |



CCA - COMPETENCE CENTRE

HTL Anichstraße



| | | |
|-----------|---|------------|
| 9.2 | IDS & IPS | 60 |
| 9.3 | Honeypot | 60 |
| 9.4 | VPN | 61 |
| 9.5 | ESA (Email Security Appliance) & WSA (Web Security Appliance) | 61 |
| 9.6 | IPsec | 62 |
| 10 | Hashfunktionen | 63 |
| I | Quellcodeverzeichnis | III |

Teil I

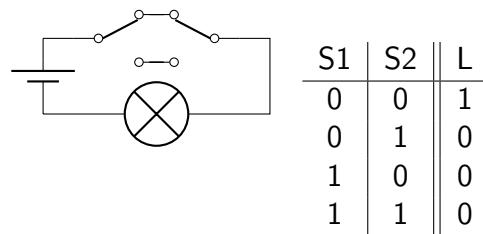
2BHWII

1 Digitaltechnik

Digital: kann nur bestimmte (diskrete) Werte annehmen (z.B. 0/1, TRUE/FALSE; Digitaluhr 08:35 → 08:26 → 08:37)

Analog: kann beliebige (kontinuierliche) Werte annehmen (Temperaturmesswerte: 38.089°C, 40.05°C).

Aufgabe: Schaltung, mit zwei Schaltern, mit denen man Licht aus- und einschalten kann



Allgemein

- mehrere Eingänge (a, b, c, d,...) (Schalter)
- einen Ausgang (x) (Lampe)

1.1 Boolesche Algebra

Grundrechenwege

- UND (konjunktion)

| a | b | $a \wedge b$ |
|---|---|--------------|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

circuit

- ODER (disjunktion)

| a | b | $a \vee b$ |
|---|---|------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

circuit

- NICHT (disjunktion)

| a | $\neg a$ oder \bar{a} |
|---|-------------------------|
| 0 | 1 |
| 0 | 0 |

$$x_1 = a \wedge (b \vee c)$$

$$z^{\text{Anzahl der Variablen}} = 2^3 = 8$$

| a | b | c | $b \vee c$ | x_1 |
|---|---|---|------------|-------|
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |

$$x_2 = a \wedge b \wedge c \wedge \bar{a}$$

| a | b | c | $\wedge a$ | $a \wedge b$ | $a \wedge b \wedge c$ | x_2 |
|---|---|---|------------|--------------|-----------------------|-------|
| 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 |

$$x_3 = a \wedge b \wedge (\bar{a} \vee c)$$

| a | b | c | \bar{a} | $(a \vee c)$ | $a \wedge b$ | x_3 |
|---|---|---|-----------|--------------|--------------|-------|
| 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 |

$x_3 = a \wedge b \wedge c$

Rechenregeln

- Vorrangsregeln
 $\neg > \wedge > \vee$
- Kommutativgesetz
 $a \wedge b = b \wedge a$
 $a \vee b = b \vee a$
- Assoziativgesetz
 $a \wedge (b \wedge c) = (a \wedge b) \wedge c$
 $a \vee (b \vee c) = (a \vee b) \vee c$
- Distributivgesetz
 $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$
 $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$
- De Morgan'sche Gesetze
 $\overline{a \wedge b} = \bar{a} \vee \bar{b}$
 $\overline{a \vee b} = \bar{a} \wedge \bar{b}$

Teil II

3BHWII

1.2 Bussysteme

Bei einem Bussystem teilen sich mehrere Teilnehmer (z.B. Arduinos) einen Übertragungsweg.

Grundlegende Eigenschaften

- Übertragungsart
 - Seriell: Zeichen werden nacheinander übertragen
 - Parallel: Zeichen werden gleichzeitig übertragen
- Topologie
 - Punkt-zu-Punkt
 - Linientopologie
 - Sterntopologie
 - Ringtopologie
 - Baumtopologie

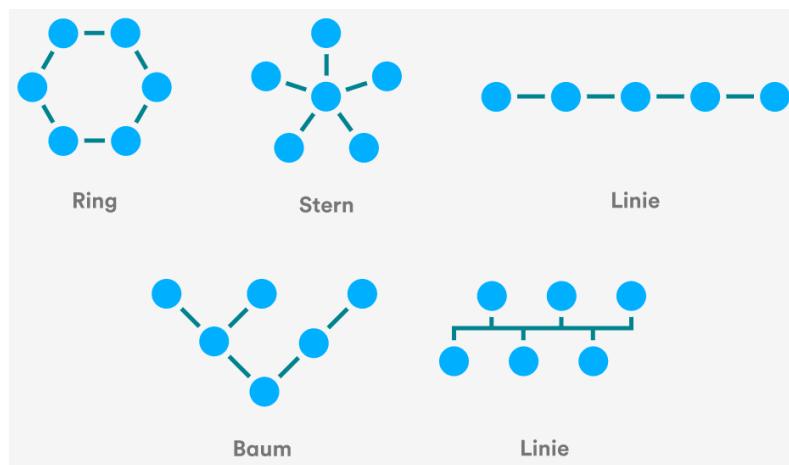


Abbildung 1.1: Bustopologiearten

- Priorisierung, Kollisionsvermeidung
- Synchronisierung
 - synchron: eigene Taktleitung
 - asynchron: keine Taktleitung, fix vorgegebene Übertragungsrate
 Einheit: $1 \text{ baud} = 1 \frac{\text{bit}}{\text{s}}$

- Fehlererkennung

Konkrete Beispiele: RS-232, CAN, I2C, SPI, USB, Bluetooth

1.2.1 RS-232 (UART)

RS-232 kann zwei Geräte miteinander verbinden.

- Topologie: Punkt-zu-Punkt
- Priorisierung, Kollisionsvermeidung: nicht nötig (2 Kabel: 1x Rx, 1x Tx)
- Synchronisierung: asynchron
- Übertragungsart: Seriell 3 bis 15V '1'
 -3 bis -15V '0'

Achtung: Alle Arduinos nutzen den gleichen GND-Pin!

- Fehlererkennung: eventuell mit einem Parity-Bit

Konkrete Umsetzung

Beim RS-232 bestehen die Daten aus einem Startbit, 5-8 Datenbits, dann eventuell ein Parity-Bit und am Ende 1 bis 2 Stopbits

Bsp: 9600 8E1

| 9600 | 8 | E | 1 |
|------------------|----------------------|---------------------|-----------------------|
| Übertragungswert | Anzahl der Datenbits | ein Even-Parity-Bit | Anzahl der Stopppbits |

'a' ASCII, 97 → 011 0000 1

| | | | |
|-------|-----------|------------|-----------|
| 0 | 01100001 | 1 | 1 |
| Start | Datenbits | Parity-bit | Stopp-Bit |

Das Parity-Bit kann

- O ... Odd
- E ... Even
- N ... None

sein

Bsp 2: 96000 8O1

'S', 83 → 1010011 0 1010011 1 1

'E', 69 → 01000101 0 1000101 0 1

UART am Arduino

Der Arduino besitzt ein UART-Bauteil, welches für die aktuelle Kommunikation genutzt

werden kann. UART-Bauteil sendet immer mit 8 Datenbits und immer mit einem Stoppenbit.

Serial Funktionen

`Serial.available()` ... wandelt eingegebene Zeichen in ASCII-Zeichen/Zahlen um
Beispiel mit Ausgabe: HELLO → 72 69 76 76 79 10

`Serial.print()` ... gibt Daten im Serial Monitor aus

z.B. `Serial.print(78)` → 78
`Serial.print("Test")` → Test

Weiters kann man Zahlen in anderen Zahlensystemen umwandeln und ausgeben (BIN, OCT, DEC, HEX)

z.B. `Serial.print(78, BIN)` → 1001110
`Serial.print(78, HEX)` → 4E

oder auch auf Nachkommastellen runden:

z.B. `Serial.print(1.23456, 0)` → 1
`Serial.print(1.23456, 2)` → 1.23
`Serial.print(1.23456, 4)` → 1.2345

`Serial.println()` ... gleich wie `Serial.print()`, jedoch beginnt es mit ASCII 13 oder '\r' (neue Zeile beginnen) und endet mit ASCII 10 oder '\n' (Zeil beenden).

`Serial.read()` ... gleich wie `Serial.available()`

`Serial.write()` ... ähnlich wie `Serial.print()` und `Serial.available()` jedoch wird hier ASCII-Zahlen in Buchstaben umgewandelt.

z.B. `Serial.write(72)` → H
`Serial.write(69)` → E
`Serial.write(76)` → L
`Serial.write(76)` → L
`Serial.write(79)` → O
`Serial.write(32)` → SPACE

1.2.2 CAN-Bus (Controll Area Network)

Der CAN-Bus ist ein serieller Bus der speziell für die Automobilindustrie entwickelt wurde

Aktuelle Anwendungen

- Autos (Vernetzung von Steuergeräten und Sensoren)



- Flugzeuge
- Raumfahrt
- Medizintechnik
- ...

Der CAN-Bus arbeitet nach dem Multi-Master-Prinzip und bildet eine Linientopologie.

Aufbau

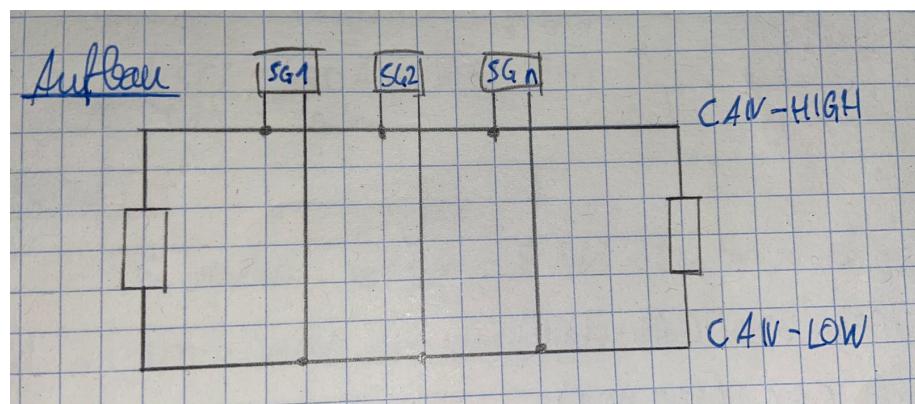


Abbildung 1.2: CAN-Bus Aufbau

Der CAN-Bus ist ein asynchroner Bus

| | | | | | |
|-----------------|----------|---|---|----------|----|
| Datenwert | Startbit | 1 | ✓ | Startbit | 1 |
| Message ID | 11 | ← | | Daten | 8 |
| Steuerbits | 7 | ✓ | | (Parity | 1) |
| Datenbits | 0-64 | ✓ | | Stopp | 1 |
| Fehlererkennung | 15 | ← | | | |
| Steuerbits | 3 | ✓ | | | |
| Stopbits | 7 | ✓ | | | |

Message ID

Dort steht um welche Art von Nachricht es sich handelt und wie wichtig die Nachricht ist (z.B. Wasser, Öl, Airbag). Desto kleiner die Message ID ist, desto wichtiger ist die Nachricht. Jedes Sg kann nur eine Message ID aussenden (Priorisierung).

Kollisionsvermeidung CSMA/CR

SG1: M-ID = 000100

SG2: M-ID = 011011

SG3: M-ID = 001101

0 ... dominant

1 ... rezessiv

Sicherheit

- zwei Leitungen (CAN-LOW, CAN-HIGH)
- Fehlererkennung mit CRC (15 Bit)
- Stuffbit (Stopfbit) Bei fünf gleichen Bits wird ein anderes eingefügt
0000000
00000100

Zusammenfassung

- seriell
- Message ID mit CSMA/CR
- Linientopologie
- asynchron
- Stuffbit, CRC, 2 Leitungen

1.2.3 I2C-Bus (IIC, I²C)

Der I2C-Bus wurde Anfang 80er Jahre von der Firma Philips entwickelt.

Der I2C-Bus ...

- ... ist seriell
- ... hat 2 Leiter: SCL & SDA (Serial Clock, Serial Data)
- ... ist synchron
- ... funktioniert nach Master-Slave-Prinzip.
Es gibt einen Master (es wären mehrere möglich) → keine Kollisionen, Priorisierung unnötig durch Master

Aufbau

- Linientopologie
- 2 Leitungen mit Pullup-Wiederstände → Ruhe zustand HIGH (SCL Takt, SDA Daten; A5, A4)

Ablauf

- es werden immer 8 Bit Datenwerte gesendet
- um Daten zu senden muss der Pegel der Datenleitung stabil sein (0 oder 1), falls die Takteleitung auf HIGH ist. Sobald die Takteleitung auf LOW ist, kann die Datenleitung das Bit setzen.

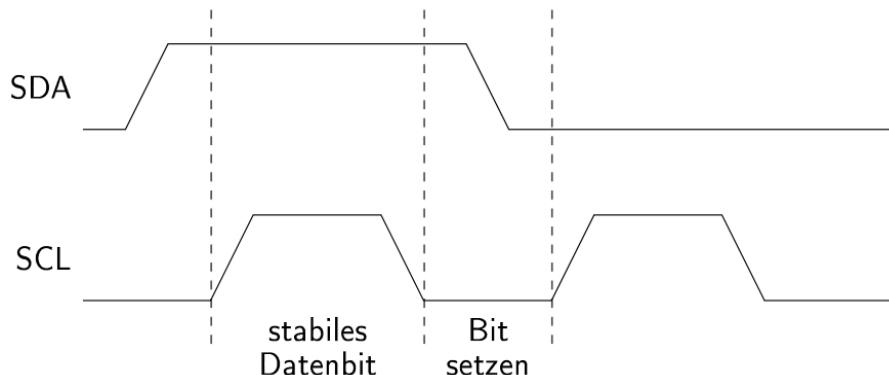


Abbildung 1.3: I2C-Bus Bitsetzung

- Steuersignale
 - Startsignal (fallende Flanke, während der Takt auf HIGH ist)
 - Stopsignal (steigende Flanke, während der Takt auf HIGH ist)

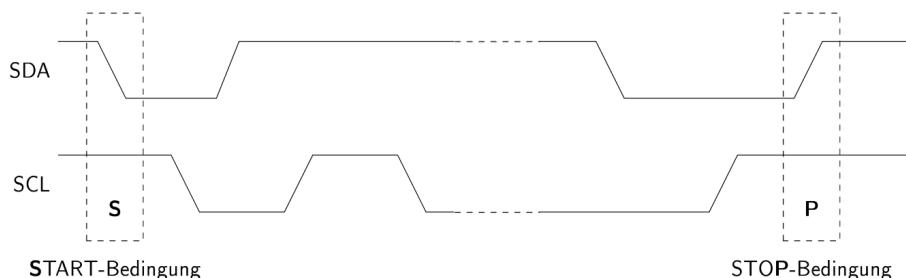


Abbildung 1.4: I2C-Bus Steuersignale

Adressierung

- 7 Bit-Adressen für die Slaves
 → 128 mögliche Teilnehmer (eig. 112)
- 0000000 → general call address (Broadcastadresse)
- Das 8te Bit sagt ob der Master lesen oder schreiben soll

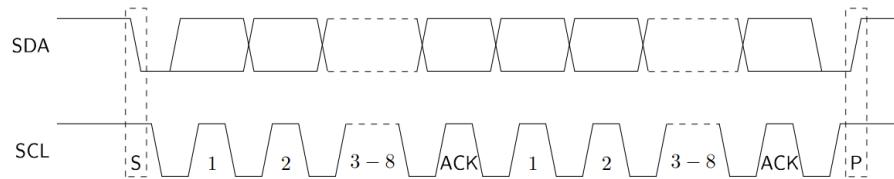


Abbildung 1.5: I2C Ablauf

Zusammenfassung

- seriell
- Linientopologie
- Master-Slave → unnötig (bei einem Master)
- synchron
- ACK, Takt

| Master | Slave |
|--|---|
| <pre>Wire.beginTransmission(address); Wire.endTransmission(); Wire.read(); Wire.write(); Wire.available();</pre> | <pre>Wire.begin(address); Wire.onReceive(); Wire.onRequest();</pre> |

I2C-Display (OLED)

Bibliotheken installieren: GFX, SSD 1306 (Adafruit), BusIO

Wire → I2C Scanner (Adresse auslesen)

SSD 1306 → 128x64 I2C

OOP am Arduino (C++)

Objekt besteht aus

- Attribute, Felder, Eigenschaften
- Konstruktor
- Funktionen/Methoden



2 Finite State Machine (Endlicher Automat)

Die Finite State Machine ist ein Modell eines Verhaltens. Es ist ein graphischer Entwurf, der flexibel Erweiterbar ist, sowie ein Programmierkonzept.

Beispiel Ampel

1. Entwurf eines Zustanddiagrammes mit Übergang
2. Zustand durchnummerieren (wird global gespeichert)
3. Übergänge sind if-Bedingungen (Ausnahme: sofortige Aktionen)
4. Zustände sind Methoden: Zeitsensible Übergänge benötigen 2 Zustände (starten und warten)

Ampel: 3s rot, 5s grün

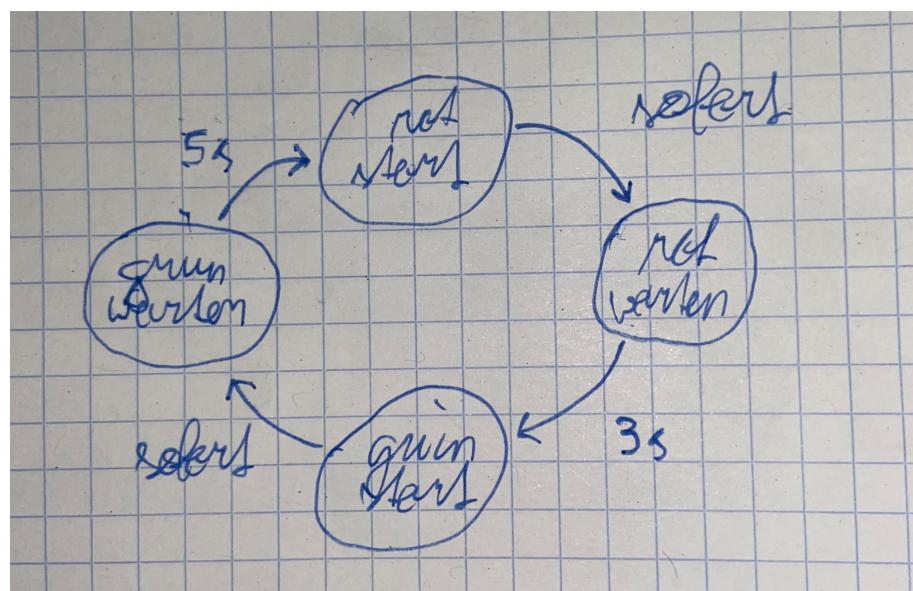


Abbildung 2.1: Finite State Machine: Ampel

(anderes Bsp im Heft: Baustellampel, Totmannschaltung)

Teil III

4BHWII

3 IPv4 (Internet Protocol Version 4):

Eine IPv4-Adresse ist eine 32-Bit Zahl. Es gibt also $2^{32} \approx 4,3$ Milliarden IPv4-Adressen.

Bsp:

| | | | |
|-----------------|----------|----------|----------|
| 11000000 | 10101000 | 00001010 | 00001010 |
| 192 | 168 | 10 | 10 |
| → 192.168.10.10 | | | |

Schreibweise:

Die IPv4-Adresse wird in Dotted Decimal Notation geschrieben. Die IP-Adresse wird in 8-Bit Blöcke (Oktetten) geteilt, dezimal übersetzt und durch Punkte getrennt.

Verwendung

Jedes Gerät soll durch eine Adresse (IP-Adresse) eindeutig identifiziert werden. Zusätzlich sollten auch Gruppen (Netze) von Computern erstellt werden (mit Subnetzmasken). Ein Gerät mit IP-Adresse nennt man Host.

Subnetmask

Ist eine 32-Bit Zahl, die in Dotted Decimal Notation beschrieben wird. Es kommen zuerst alles Einsen und nach der ersten Null nur noch Nullen.

Typische Subnetmasken:

| | Präfix | Hosts |
|---------------|--------|---------------------------|
| 255.0.0.0 | 8 | $2^{24} - 2 = 16.777.214$ |
| 255.255.0.0 | 16 | $2^{16} - 2 = 65.534$ |
| 255.255.255.0 | 24 | $2^8 - 2 = 254$ |

Bsp:

| | | |
|----------------|-------------|-------------------|
| Telefonnummer | IP-Adresse | |
| +43 664 123456 | 172.16. | 20.25 |
| Netz | einzigartig | 255.255. 0.0 |
| | | Netzteil Hostteil |

Die Subnetzmaske trennt die IP-Adresse in Netzteil und Hostteil. IP-Adresse und Subnetzmaske gehören immer zusammen.

1) 2 IP-Adressen im gleichen Netz

10.10.226.120 / 24

10.10.226.80 / 24

2) 2 IP-Adressen nicht im gleichen Netz

11.40.30.124 / 24

14.8.50.100 / 24

3) Anzahl der Hosts

$2^8 - 2$ 10.10.226.0 (Netzadresse)

10.10.226.255 (Broadcastadresse)

192.168.20.100 / 8

Netz: 192.0.0.0

Broad: 192.255.255.255

4 Netzwerke im Alltag und Grundbegriffe

Netzwerk Komponenten

- Endgeräte (PC, Handy, Uhr, TV, Server,...)
- Intermediary Devices (Router, Repeater, Switch, Hub, Access Point)
- Übertragungsmedien (Drahtlos, Kupfer, Glasfaser)

Host-Aufgaben

- Client-Server-Modell
- Peer-to-Peer Modell
 - + Komplexität
 - + Leichter zum Aufsetzen
 - Security
 - Erweiterbarkeit

Netzwerk Dokumentation

- Physische Topologie (Räume, ...)
- Logische Topologie (Netze, ...)

Netzwerke nach Größe

- SOHO ... small office home office
- LAN ... local area network
- MAN ... metropolitan area network
- WAN ... wide area network
- Internet

Netzwerke nach Funktion

- SAN ... storage area network
- Intranet, Extranet

Internetzugang

- Kabel (Glasfaser)
- DSL / Dial Up
- Mobilfunknetz
- Satellit

Trends

- Video / Streaming
- Cloud
- Drahtlos (5G)
- BYOD (bring your own device)
- Online Collaboration
- Powerline Method

Netzwerkarchitektur

- Quality of Service QoS
- Erweiterbarkeit
- Security
- Fehlertoleranz

Security

- Ransomware
- DoS / DDoS
- Virus, Wurm, Trojaner
- Social Engineering
- Zero-Day-Attack

4.1 Referenzmodell (OSI und TCP/IP)

| OSI | | Protokolle | TCP/IP |
|-----|--------------------|--------------------------|----------------------|
| 7 | Application Layer | HTTPS, FTP, Telnet, SSH | Application Layer |
| 6 | Presentation Layer | POP, SMTP, IMAP | |
| 5 | Session Layer | DHCP, NTP, DNS | |
| 4 | Transport Layer | TCP, UDP | |
| 3 | Netzwerk Layer | IP, ICMP; OSPF, BGP, RIP | |
| 2 | Data Link Layer | Wifi, Ethernet, ARP | |
| 1 | Physical Layer | | Network Access Layer |

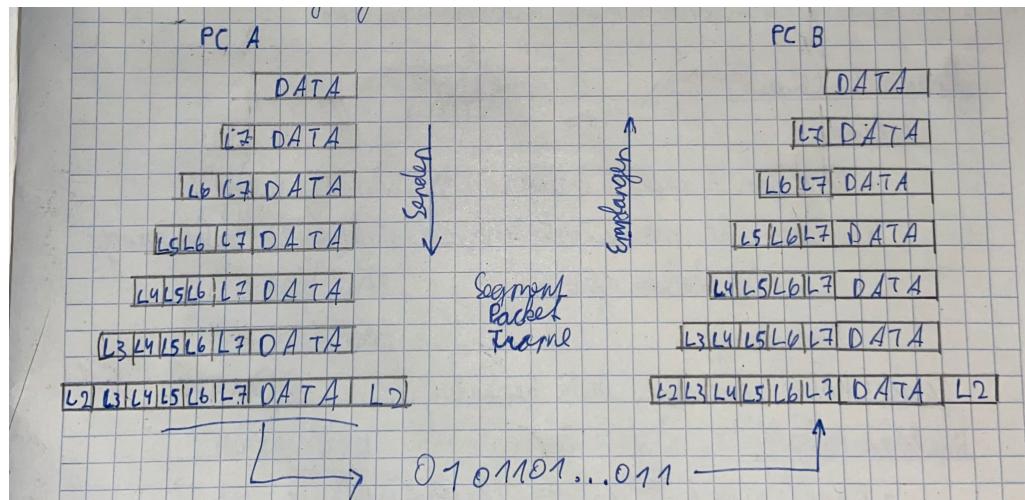


Abbildung 4.1: OSI-Modell Datenübertragung

Layer 1 (Physical): Bits übertragen

Layer 2 (Data Link): Lokale Adressierung, Fehlererkennung

Layer 3 (Network): Globale Adressierung, Routing

Layer 4 (Transport): Datenpaketazuordnung, Segmentierung, Datenfluss steuern

Layer 5 (Session): Session Verwalten, Verschlüsselung

Layer 6 (Presentation): Darstellung der Daten

Layer 7 (Application): Funktionen für die Application

Cisco CLI

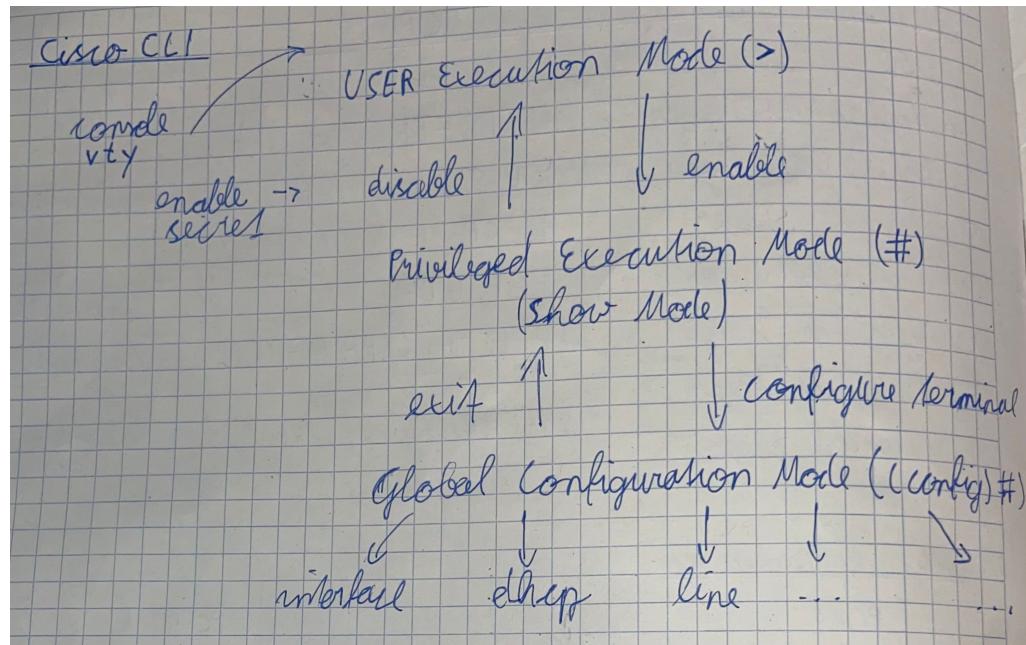


Abbildung 4.2: Cisco CLI

4.1.1 Layer 1 (Physical)

Aufgaben

- Bits von A nach B bringen
- elektrische, mechanische oder andere physische Verbindung zwischen zwei Geräten
- Kodierung

Geräte: Kabel, Antenne, Hub, Repeater,...

Wichtige Begriffe

- Bandbreite (bits/s → theoretisch)
- Durchsatz (bits/s → praktisch)
- Latenz (Dauer der Daten von A bis B in ms)

Typische Medien

- Kupferkabel (Twisted-Pair-Kabel)
 - + Günstig
 - + einfache Handhabung
 - ≈ Distanz (ca 100m)
 - ≈ Geschwindigkeit
 - Interferenzen (Störungen)

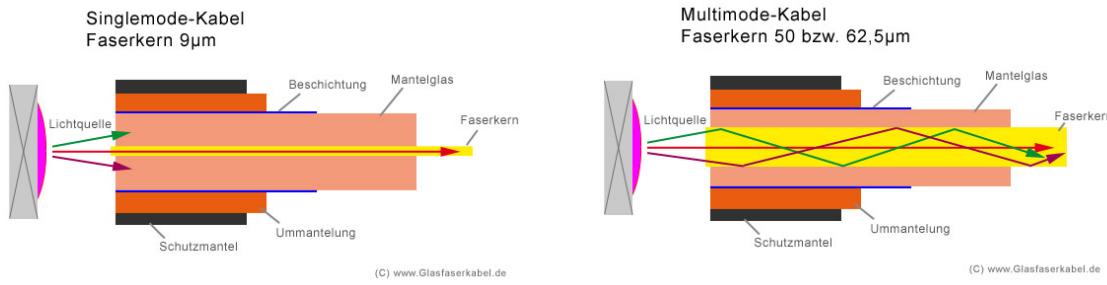
Straight Through (beide Enden gleich)

Crossover (verschiedene Enden)

(durch Auto MDIX werden Enden automatisch konfiguriert)

- Koaxialkabel
- Glasfaserkabel

Arten: Single-Mode (Senden Laser, Reichweite 1-10km)
Multi-Mode (Senden LED, Reichweite ca 600m)



(a) Single-Mode

(b) Multi-Mode

Abbildung 4.3: Glasfaserkabelarten

- | | |
|--|-------------------------|
| + Speed + Reichweite + Störungen | - Teuer - Handhabung |
|--|-------------------------|

- Drahtlos

Übertragung: elektromagnetische Wellen über Luft

- | |
|---|
| + Flexibel - Störungen - Shared Medium - Reichweite (ca 100m), Hindernisse - Security |
|---|

4.1.2 Layer 2 (Data Link)

Aufgaben

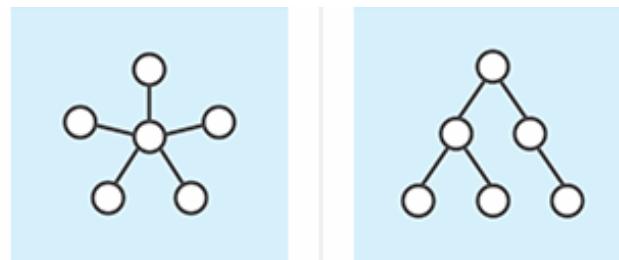
- lokale Adressierung
- Fehlererkennung
- Zugang zum Medium herstellen
- Kommunikation mit Layer 3

Geräte: Netzwerkkarte, Switch, Bridge,...

Standards: Wifi (802.11), Ethernet (802.2, 802.3)

Topologie

- Sterntopologie
- Baumtopologie
- Punkt-zu-Punkt



| | |
|--|--|
| <p>Stern Verfügt über ein zentrales Gerät, das Daten an andere Knoten im System überträgt.</p> | <p>Baum Verbindet Geräte in einer Struktur, die einem Baum ähnelt, bei dem übergeordnete Knoten mit untergeordneten Knoten verbunden sind.</p> |
|--|--|

Abbildung 4.4: Baum- und Stern topologie

Ethernet

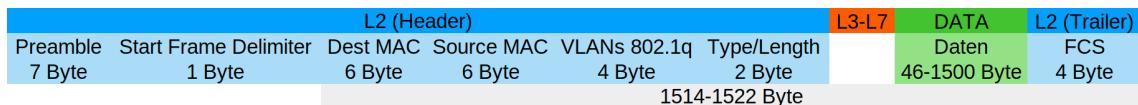


Abbildung 4.5: Ethernet Frame

MAC-Adresse

Die MAC-Adresse ist eine 48-Bit Zahl und wird in hexadecimal dargestellt.

Bsp:

Hersteller für den Hersteller einzigartig

DC F5 05 |17 9A 69

Jede Netzwerkkarte besitzt eine weltweit einzigartige (theoretisch) MAC-Adresse.

Type

Kodierung für Layer 3

0x800 → IP

0x806 → ARP

Fehlerkennung

Frame Checksum (CRC)

Polynomdivision mit einem Polynom von Grad 32

Funktion eines Switches

Der Switch baut mit der Source-MAC seine MAC-Tabelle auf. Dort steht zu jeder MAC-Adresse der passende Port. Falls die MAC-Adresse schon eingetragen ist, wird ein Timer aktualisiert. Sollte es noch keinen Eintrag geben wird er hinzugefügt und bleibt dort eine gewisse Zeit (5 Minuten) bevor er gelöscht wird. Der Switch vergleicht die Destination-MAC mit seiner MAC-Tabelle. Falls der Switch keinen Eintrag findet sendet er an alle Ports (Flooding, Unknown Unicast). Sonst sendet er an den Port, wo er den Frame bekommen hat.

Layer 2 Broadcast Adresse: FF:FF:FF:FF:FF:FF

L2, L3 Adressierung

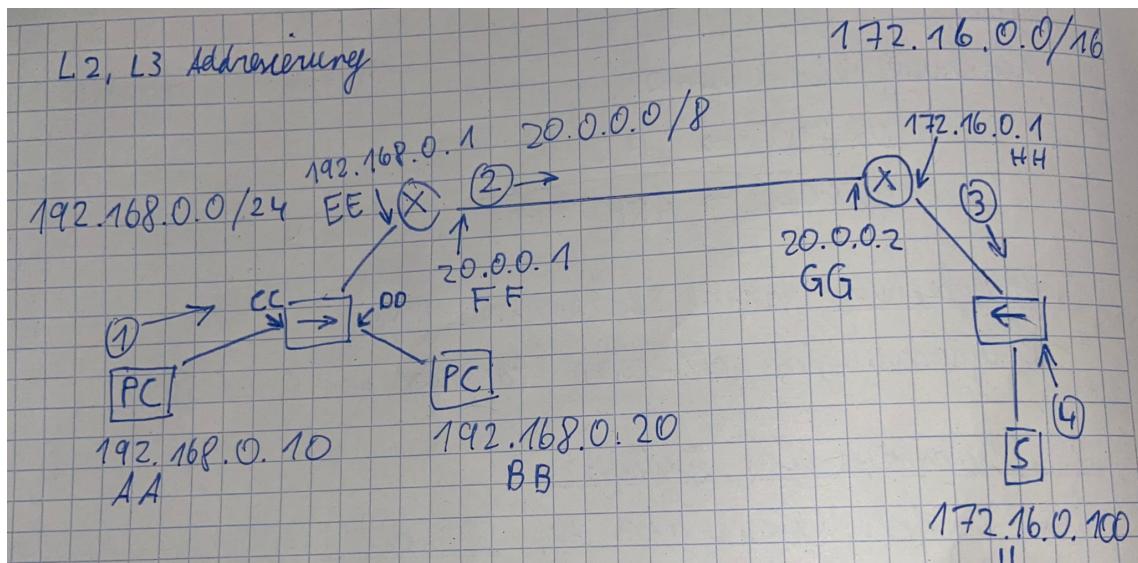


Abbildung 4.6: Layer 2 & 3 Adressierung

| | Source MAC | Destination MAC | Source IP | Destination IP |
|---|------------|-----------------|--------------|----------------|
| 1 | AA | EE | 192.168.0.10 | 172.16.0.100 |
| 2 | FF | GG | 192.168.0.10 | 172.16.0.100 |
| 3 | HH | II | 192.168.0.10 | 172.16.0.100 |
| 4 | II | HH | 172.16.0.100 | 192.168.0.10 |

ARP (Address Resolution Protocol)

Nutzt ein Host um zu einer gegebenen IP-Adresse die passende MAC-Adresse zu finden

ARP-Request (Broadcast)

Source MAC: eigene MAC-Adresse

Destination MAC: FF-FF-FF-FF-FF-FF

Type: 0x806 Danach ARP-Header (IP, MAC, Protokoll)

ARP-Reply Unicast (auch als Broadcast möglich)

Source MAC: eigene MAC-Adresse (gesucht)

Destination MAC: MAC-Adresse (Anfrage)

Type: 0x806

Danach ARP-Header

ARP-Cache

Die Einträge werden im ARP-Cache gespeichert (ca 5 min)

IP MAC Time

ARP-Spoofing

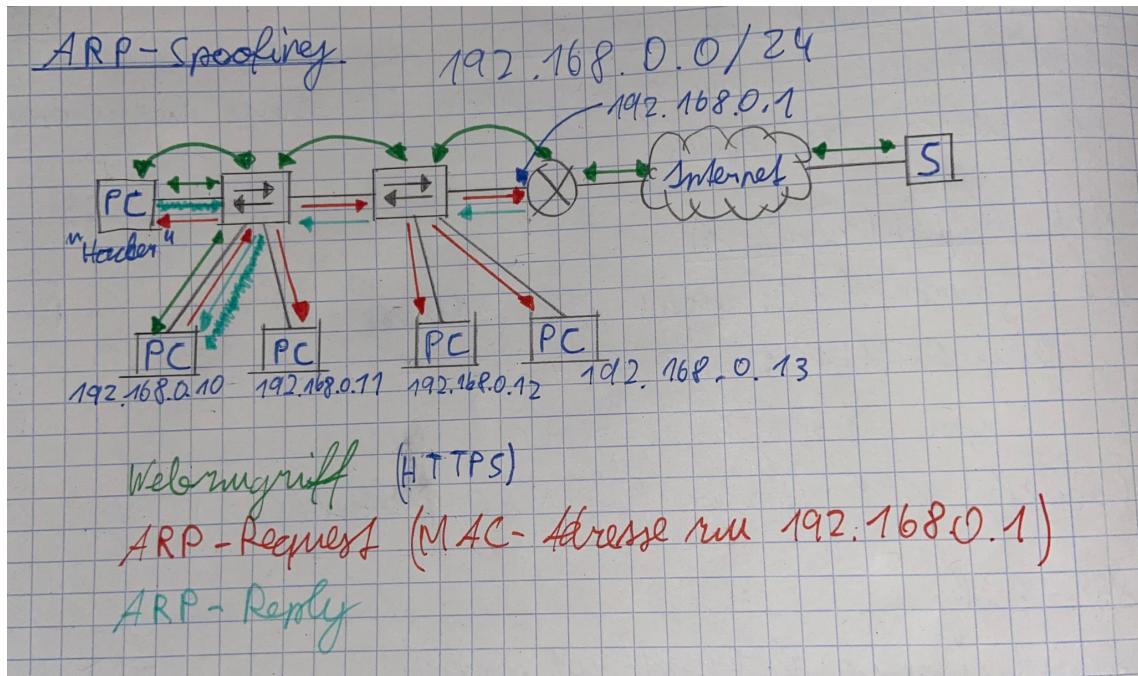


Abbildung 4.7: ARP-Spoofing

4.1.3 Layer 3 (Network)

Aufgaben

- Routing
- Globale Adressierung
- Kommunikation mit L2 & L4

Protokolle: IPv4, IPv6, ICMP, RIP, OSPF, EIGRP, IS-IS, BGP

IPv4

Eigenschaften von IP

- Verbindungslos
- Best Effort
- Medium unabhängig

IP-Header (8.2.2) Wichtige Felder: Source & Destination IP, Time-to-Live

Kommunikationsart

- Unicast (IP des Host)
- Multicast (224.0.0.0 - 239.255.255.255)
- Broadcast (letzte IP im Netz, 255.255.255.255)

Spezielle IP-Adressen

- 127.0.0.0 / 8 ... localhost
- 10.0.0.0 / 8
- 172.16.0.0 / 12
- 192.168.0.0 / 16 ... private IP-Adressen (NAT)
- 169.254.0.0 / 16 ... APIPA
- 192.0.2.0 / 24 ... Testnetz

Fazit: Zu wenig IPv4-Adressen!

Deshalb

- VLSM (variable length subnet mask)
- NAT

- IPv6

Classful Addressing (uralt)

Das erste Oktett bestimmt die Subnetzmaske (/8, /16, /24)

| | | | |
|----------|---------|------------|------------------------|
| Klasse A | 0-127 | (0...) | /8 |
| Klasse B | 128-191 | (10...) | /16 |
| Klasse C | 192-223 | (110...) | /24 |
| Klasse D | 224-239 | (1110...) | Multicast |
| Klasse E | 240-255 | (11110...) | für spätere Verwendung |

Classless Addressing (veraltet!)

Die Subnetzmasken /8, /16, /24 können beliebig verwendet werden

CIDR (Classless Inter-Domain Routing)

Es können beliebige Subnetzmasken (z.B. /25, /26, ...) verwendet werden. Alle Subnetze werden gleich groß.

VLSM (variable length subnet mask)

Alle Subnetzmasken können beliebig verwendet werden. Die Netzte dürfen sich nicht überschneiden.

Subnetzmasken

| Präfix Notation | Dotted Decimal Notation | Hosts | Subnetz von /24 |
|-----------------|-------------------------|----------------------|-------------------------|
| /25 | 255.255.255.128 | $2^7 - 2 = 126$ | 2 |
| /26 | 255.255.255.192 | $2^6 - 2 = 62$ | 4 |
| /27 | 255.255.255.224 | $2^5 - 2 = 30$ | 8 |
| /28 | 255.255.255.240 | $2^4 - 2 = 14$ | 16 |
| /29 | 255.255.255.248 | $2^3 - 2 = 6$ | 32 |
| /30 | 255.255.255.252 | $2^2 - 2 = 2$ | 64 |
| /31 | 255.255.255.254 | $2^1 - 2 = 0$ | für spezielle Anwendung |
| /20 | 255.255.240.0 | $2^{12} - 2 = 4.094$ | / |

Bsp 1 (CIDR):

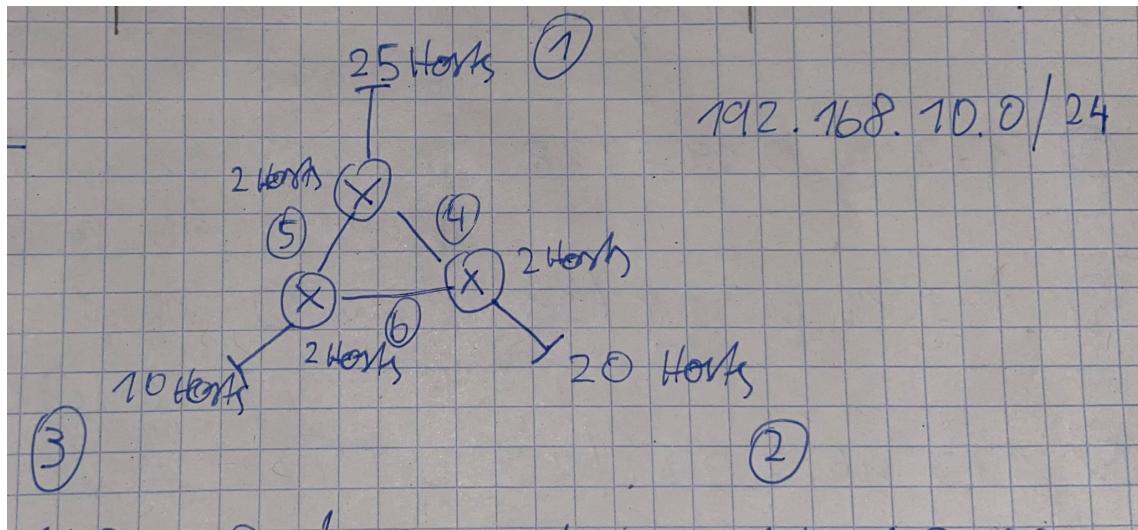


Abbildung 4.8: CIDR Beispiel

| | | | |
|---|---------------------|-------------|-----------------|
| 1 | 192.168.10.0 / 27 | Netzadresse | 192.168.100.0 |
| | | Broadcast | 192.168.100.31 |
| 2 | 192.168.10.32 / 27 | Netzadresse | 192.168.100.32 |
| | | Broadcast | 192.168.100.63 |
| 3 | 192.168.10.64 / 27 | Netzadresse | 192.168.100.64 |
| | | Broadcast | 192.168.100.95 |
| 4 | 192.168.10.96 / 27 | Netzadresse | 192.168.100.96 |
| | | Broadcast | 192.168.100.127 |
| 5 | 192.168.10.128 / 27 | Netzadresse | 192.168.100.128 |
| | | Broadcast | 192.168.100.159 |
| 6 | 192.168.10.160 / 27 | Netzadresse | 192.168.100.160 |
| | | Broadcast | 192.168.100.191 |

Bsp 2 (VLSM):

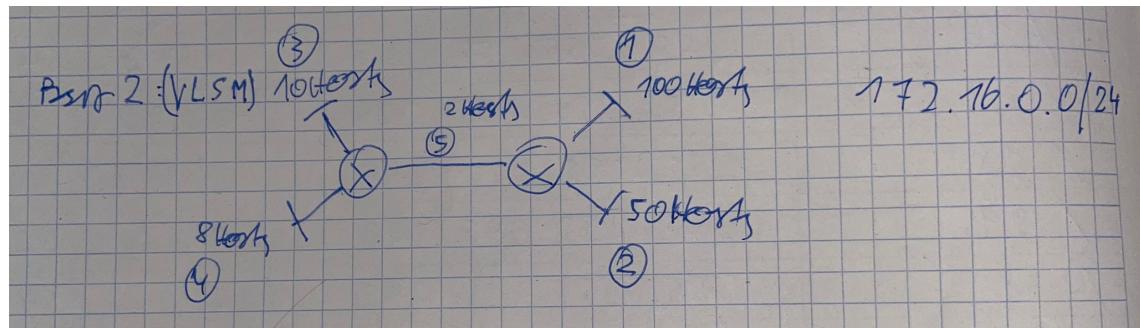


Abbildung 4.9: VLSM Beispiel

| | | | |
|---|-------------------|-------------|--------------|
| 1 | 172.16.0.0 / 25 | Netzadresse | 172.16.0.0 |
| | | Broadcast | 172.16.0.127 |
| 2 | 172.16.0.128 / 26 | Netzadresse | 172.16.0.128 |
| | | Broadcast | 172.16.0.191 |
| 3 | 172.16.0.192 / 28 | Netzadresse | 172.16.0.192 |
| | | Broadcast | 172.16.0.207 |
| 4 | 172.16.0.208 / 28 | Netzadresse | 172.16.0.208 |
| | | Broadcast | 172.16.0.223 |
| 5 | 172.16.0.224 / 30 | Netzadresse | 172.16.0.224 |
| | | Broadcast | 172.16.0.227 |

(PT: 10.4.3, 11.5.5, 11.9.3, 11.10.1)

4.1.4 Layer 4 (Transport)

Aufgaben

- Anwendungen identifizieren
- Segmentierung
- ev. Flusskontrolle, Verbindungsauflösung & abbau
- Kommunikation mit L3 & L5

Protokolle:

- TCP (Transmission Control Protocol)
- UDP (User Datagram Protocol)

| TCP | UDP |
|--|---|
| Anwendungen identifizieren (Ports) Segmentierung Verbindungen auf- bzw abbauen Segmente ordnen wiederholtes Senden Flusskontrolle | Anwendungen identifizieren (Ports) Segmentierung |

TCP: HTTP (80)/HTTPS (443), SMTP (25), POP (110), IMAP (143), Telnet (23), SSH (22), FTP (20/21),...
 UDP: DNS (53), DHCP (67/68), VoIP, Streaming,...

Ports

Der Port ist eine 16-Bit Zahl $\rightarrow 2^{16} = 65.536$

Der Port identifiziert die Anwendung, sowohl beim Server als auch beim Client.

Gruppe von Ports

| | |
|------------------|-----------------|
| Well-Known-Ports | 0 - 1.023 |
| Registered-Ports | 1.024 - 49.151 |
| Private Ports | 49.152 - 65.535 |

L4-Adressierung

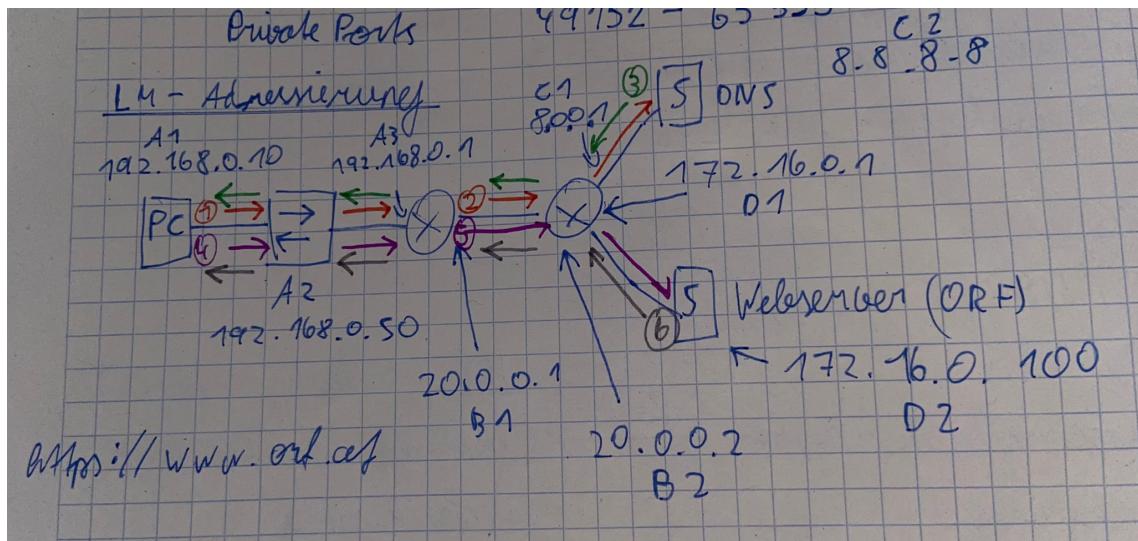


Abbildung 4.10: L4-Adressierung

| | L2 (MAC) | | L3 (IP) | | L4 (Ports) | |
|---|----------|-------------|--------------|--------------|------------|-------------|
| | Source | Destination | Source | Destination | Source | Destination |
| 1 | A1 | A3 | 192.168.0.10 | 8.8.8.8 | 53.722 | 53 |
| 2 | B1 | B2 | 192.168.0.10 | 8.8.8.8 | 53.722 | 53 |
| 3 | C2 | C1 | 8.8.8.8 | 192.168.0.10 | 53 | 53.722 |
| 4 | A1 | A3 | 192.168.0.10 | 172.16.0.100 | 60.112 | 443 |
| 5 | B1 | B2 | 192.168.0.10 | 172.16.0.100 | 60.112 | 443 |
| 6 | D2 | D1 | 172.16.0.100 | 192.168.0.10 | 443 | 60.112 |

TCP

Verbindungsaufbau: Drei-Wege-Handshake

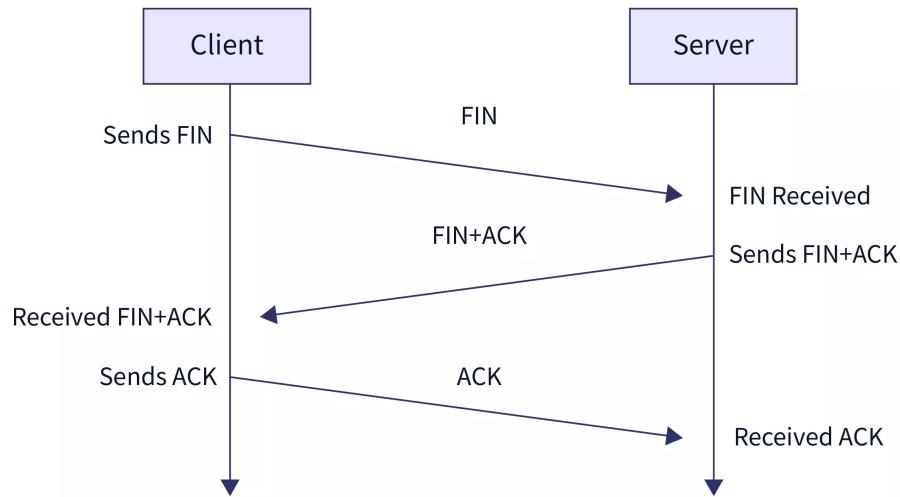


Abbildung 4.11: TCP 3-Way-Handshake

Verbindungsabbau: Zwei-Wege-Handshake

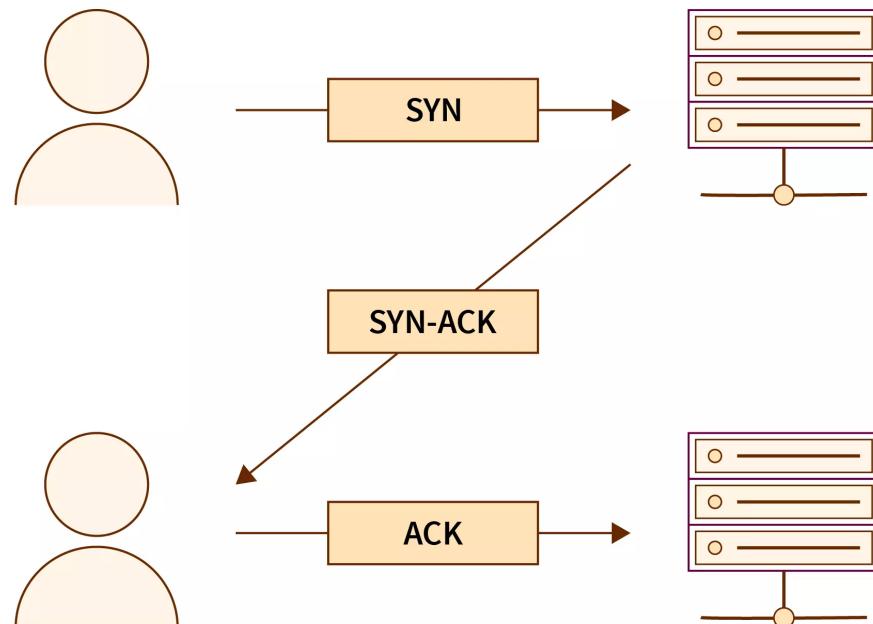


Abbildung 4.12: TCP 2-Way-Handshake

Segmentierung

Es wird eine SEQUENCENUMBER mitgeschickt. Diese gibt die Reihenfolge an. Der Client bestätigt die Segmente mit ACK-Segmenten. Die ACK-NUMBER gibt an, welches Segment als nächstes kommen soll.

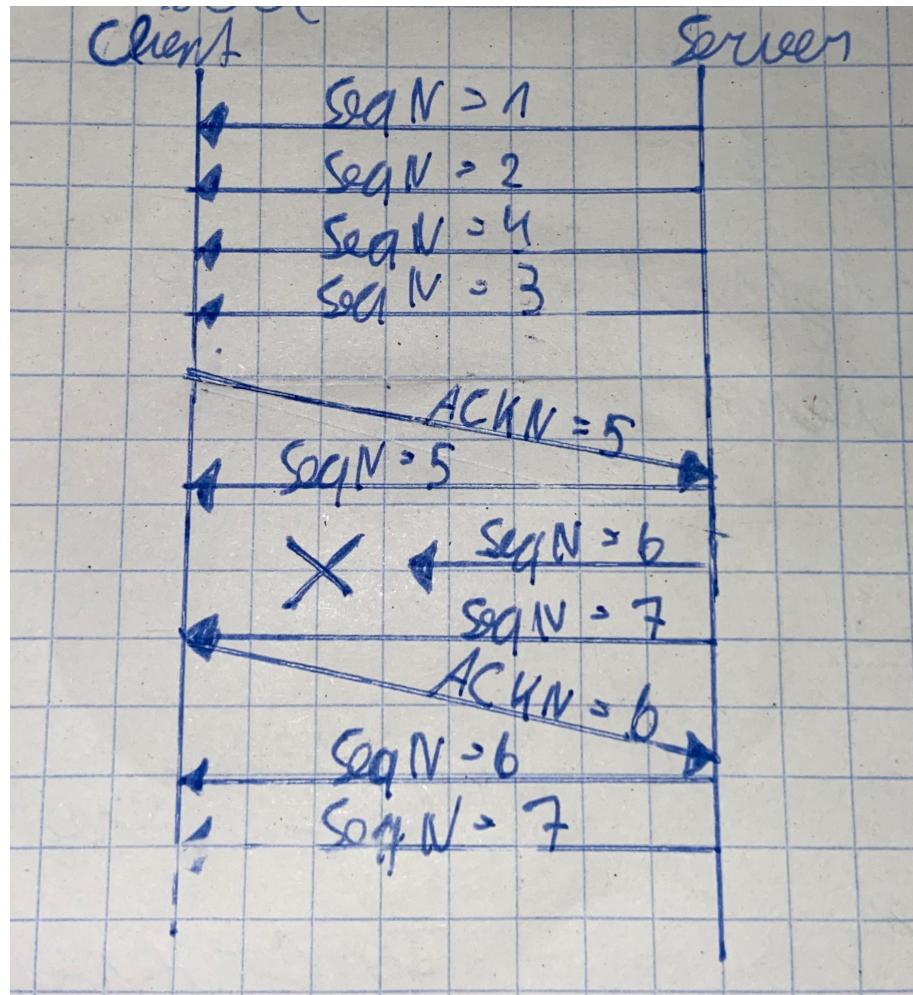


Abbildung 4.13: Layer 4 Segmentierung

Flow-Control

Die Window Size gibt an wann das nächste ACK-Segment erwartet wird.

4.1.5 Layer 5, 6, 7 (Session, Presentation, Application)

Aufgaben

- Session erstellen und halten
- Regelung der Session, Restart, Exchange, Idle
- Format und Präsentation der Daten
- Verschlüsselung und Komprimierung der Daten
- Anwendungsspezifische Informationen

Protokolle: HTTP/HTTPS, FTP, Telnet/SSH, DHCP, DNS, SMTP, POP, IMAP

DNS (Port 53, UDP)

Um zu einer Domain die passende IP-Adresse zu finden. Typische DNS-Server: 8.8.8.8

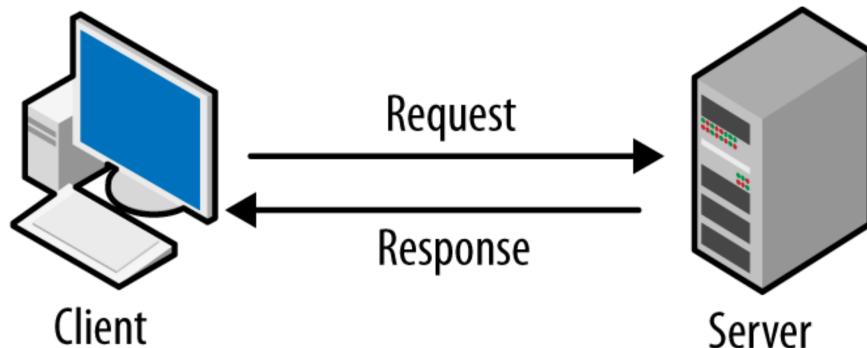


Abbildung 4.14: Request/Response Modell

Einträge

A ... IPv4-Endgerät AAAA ... IPv6-Endgerät MX ... Mail-Server

Hierarchisches DNS-Modell

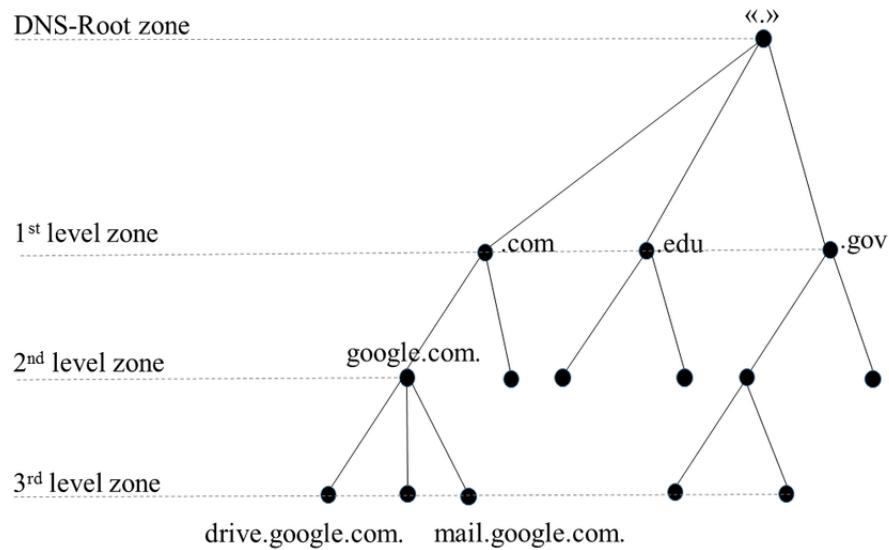


Abbildung 4.15: DNS-Hierarchie

Falls der DNS-Server keinen Eintrag findet, wird das Paket weitergeleitet. Der Client speichert die erhaltenen DNS-Einträge.

DHCP (Port 67/68, UDP)

Die Hosts erhalten dynamisch eine IP-Konfiguration (IP-Adresse, Subnetzmaske, Default Gateway, DNS-Server, Lease Time,...).

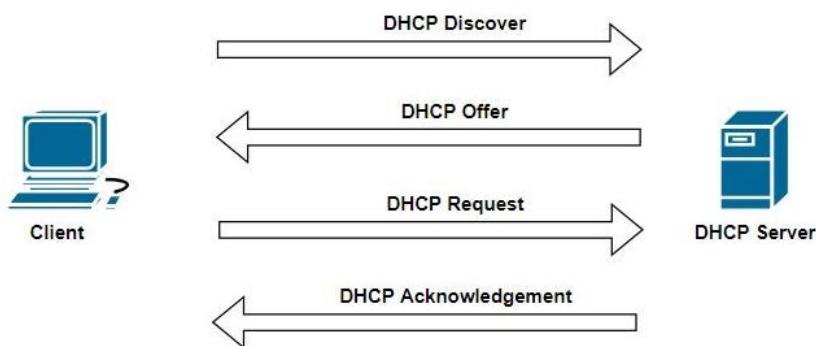


Abbildung 4.16: DHCP-Handshake

DHCP-Discover ... Broadcast

DHCP-Offer ... Unicast

DHCP-Request ... Broadcast

DHCP-ACK ... Unicast

Achtung: DHCP-Spoofing

HTTP/HTTPS (Port 80/443, TCP)

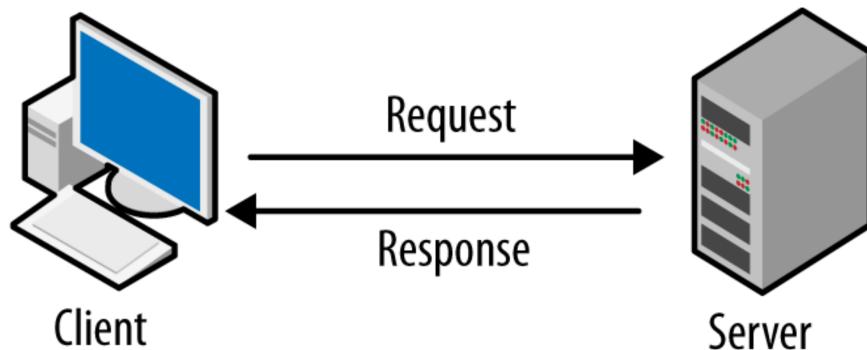


Abbildung 4.17: Request/Response Modell

| | | | |
|------|-----------|-------------------------|-----------------------|
| URL: | https:// | www.google.com/ | index.html |
| | Protokoll | Domain IP-Adresse (DNS) | Ordnerstruktur, Datei |

Befehle

Get, Post, Put, Delete,...

Bei HTTP ist alles im Klartext.

Bei HTTPS wird zusätzlich mit SSL/TLS verschlüsselt.

E-Mail

| | | | |
|-----------------|--------------|---|-----------|
| E-Mail-Adresse: | name | @ | gmail.com |
| | Benutzername | | Domain |

SMTP (Port 25, TCP)

Senden von Emails. Wird zum Senden von Mails und dem Weiterleiten zum Zielserver benutzt. SMTP kann zusätzlich Feedback geben (z.B. Ziel nicht erreichbar,...).

POP (Port 110, TCP)

Empfangen von E-Mails. Man erhält vom Server das Original. Die Mail wird am Server gelöscht (Vorteil: Speicherplatz, Security).

IMAP (Port 143, TCP)

Empfangen von E-Mails. Man erhält vom Server eine Kopie. Das Original bleibt am Server gespeichert (Vorteil: Verbindung mit mehreren Geräten ist praktisch, Backup).

4.2 VLANs

Ein physisches Netz wird in mehrere logische Teilnetze (Layer 2) unterteilt.

| Vorteile | Nachteile |
|---------------------------|-----------------|
| Kosten | (Konfiguration) |
| Security | |
| Flusskontrolle | |
| Übersicht | |
| kleinere Broadcast-Domain | |
| Effizienz & Performance | |

Arten von VLANs

- Daten VLANs
- Default VLAN (bei cisco 1)
- Voice VLAN
- Management VLAN
- Native VLAN (Frames ohne VLAN-Tag kommen in das Native VLAN, kann nur am Trunk passieren)

Access Ports transportieren nur ein VLAN.

Trunk Ports können viele VLANs transportieren.

Die VLAN-Namen werden zusätzlich im Header eingetragen (802.1q → Ethernet)

ACL

Je nach IP-Adresse (Standard) bzw. Port (Extended) wird ein Packet blockiert oder zugelassen.

Wildcardmask

Subnetzmaske: Teilt IP-Adresse in Netz- und Hostteil

1 ... Netzteil (relevant für das Netz)

0 ... Hostteil (irrelevant für das Netz)

→ unflexibel

Wildcardmaske '1' & '0' können beliebig gezählt werden

0 ... relevantes Bit der IP-Adresse

1 ... nicht relevantes Bit der IP-Adresse

255.255.255.255 any

0.0.0.0 host

Position der Wildcardmask

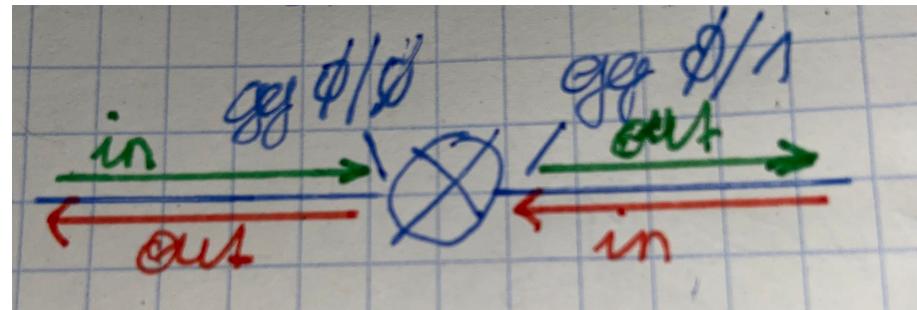


Abbildung 4.18: Position der Wildcardmask

Regeln bei Interfaces: eingehend & ausgehend

Achtung Die letzte Zeile in jeder ACL ist 'deny any'.

Static NAT (1:1 Mapping)

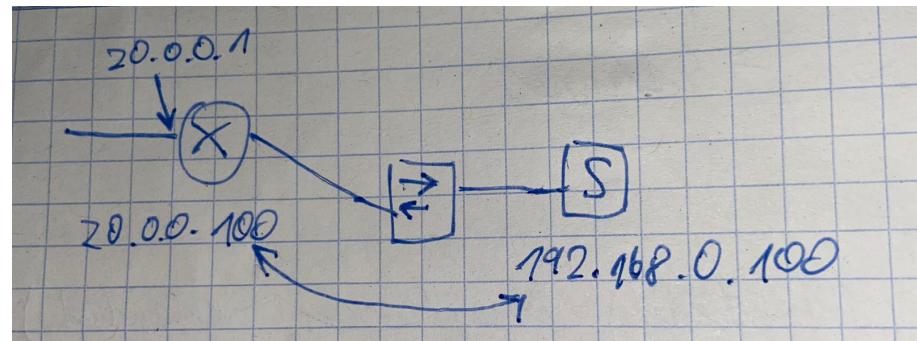


Abbildung 4.19: Static NAT, 1:1 Mapping

NAT mit PAT (n:1 Mapping)

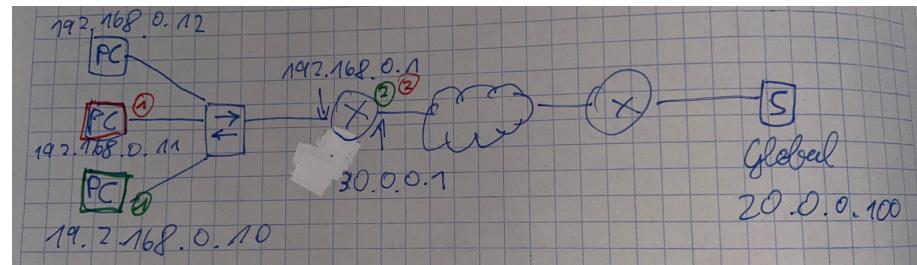


Abbildung 4.20: NAT mit PAT, n:1 Mapping

| | Source IP | Destination IP | Source Port | Destination Port |
|---|--------------|----------------|-------------|------------------|
| 1 | 192.168.0.10 | 20.0.0.100 | 51000 | 443 |
| 1 | 192.168.0.11 | 20.0.0.100 | 51000 | 443 |
| 2 | 30.0.0.1 | 20.0.0.100 | 51000 | 443 |
| 2 | 30.0.0.1 | 20.0.0.100 | 51001 | 443 |

| Vorteile | Nachteile |
|--|--|
| <ul style="list-style-type: none"> + IP-Adressen sparen + Security + IP-Adressen Schema kann frei gewählt werden | <ul style="list-style-type: none"> - Ende zw. Ende Verbindung geht verloren - Paketverfolgung und Troubleshooting - Performance |

Teil IV

5BHWII

5 Routing

Router muss entscheiden welcher Weg der 'beste' Weg ist.

→ bei welchen Interface (Netz) rauschicken = Routing

Routing Tabelle wird durch ...

- dynamisch (Routingprotokolle)
- statische Einträge

... aufgebaut (in der Praxis meist aus Mischung).

Router wählt Route mit am meisten Bits bei Ziel übereinstimmung (Vergleich von Route & Destination IP).

1) Einträge in Routing Tabelle

- Direkt verbundene Netze: Aktive & angeschlossene Netze am Router mit IP-Konfiguration → automatisch (Status Code: C, L)
- Remote Netze: statisch oder dynamisch (vom Routingprotokoll abhängig) einge-tragen (Status Code: S, R, O, E,...)
- Default Route (gateway of last resort): Next Hop falls der Router keine passende Route findet, statisch oder dynamisch.
Route: 0.0.0.0 / 0 ... 0 Bits müssen übereinstimmen

2) Eintrag in Cisco CLI

| | | | | | |
|--------|-------------|-----------|----------------------------|-------------|---------------|
| R | 30.0.4.0/24 | [120/7] | via 10.0.3.2 | 00:13:29 | Serial 10/1/1 |
| Status | Ziel | AD/Metrik | IP (ausgehendes Interface) | Zeitstempel | Interface |

Status Code

C ... connected Direkt verbundene Netzte
 L ... local IP vom Interface, lokale Route
 S ... static statisch eingegebene Route
 R ... RIP entsprechendes Routingprotokoll

- ... OSPF entsprechendes Routingprotokoll
- ... EIGRP entsprechendes Routingprotokoll

Ziel

IP-Adresse des Zielnetzes mit Präfix (nicht unbedingt Subnetzmaske). Es müssen die angegebene Anzahl von Bits (Präfix) mit Destination IP-Adresse übereinstimmen (damit Route in Frage kommt). Route mit am meisten übereinstimmenden Bits (von links). Problem: es wird keine Subnetzmaske der Destination IP mitgeschickt → normalerweise auch nicht bekannt.

Dest IP: 172.16.0.10 (letztes Oktett: 00001010)

-
- 1) 172.16.0.0 /16 Bis Bit 16 übereinstimmend
 - 2) 172.16.0.0 /24 Bis Bit 24 übereinstimmend
 - 3) 172.16.0.0 /26 Bis Bit 26 übereinstimmend
 - 4) 172.16.0.0 /30 Bis Bit 29 nicht übereinstimmend
 - 5) 172.17.0.0 /24 am 2. Oktett stimmt es nicht überein

Router wählt 3. Variante (???). Dort stimmt die angegebene Anzahl an Bits (Präfix) überein

AD: Administrative Distanz

Router kann Route über mehrere Arten lernen (z.B. statisch, RIP, OSPF,...). AD gibt an wie 'vertrauenswürdig' eine Route ist. Router verwendet Route mit niedrigster AD, andere Routen sind Backups und werden vorerst nicht im Routing Table angezeigt.
→ wenn 'beste' Route ausfällt wird nächst beste verwendet

Standard Werte bei Cisco Routern

| AD | |
|------------------|-----|
| Direkte Routen | 0 |
| Statische Routen | 1 |
| EIGRP | 90 |
| OSPF | 110 |
| RIP | 120 |

Metrik

Von einem Routingprotokoll kann der Router mehrere Routen zum gleichen Ziel lernen. Die Metrik gibt an, 'wie weit' das Ziel entfernt ist. Der Router verwendet die Route mit der geringsten Metrik. Falls eine Route ausfällt, wird auf Backup-Routen zurückgegriffen.

3) Statische Routen

Werden in kleineren Netzen mit geringen Veränderungen, bei speziellen Zielnetzen oder Router mit nur einen Nachbar (Stub-Network) verwendet.

Problem: Statische Routen werden nicht automatisch aktualisiert und müssen händisch aktualisiert werden.

4) Dynamische Routingprotokolle

Je nach Ablauf des Routingprotokolls unterscheidet man unterschiedliche Kategorien.

- Pfadvektorprotokolle

Diese Protokolle speichern den Pfad/Weg zum Ziel. Sie sind besonders effizient gegen Routing-Schleifen und eignen sich dadurch zum Routen von autonomen Systemen.

Beispiel: BGP (Border Gateway Protocol), Metrik: Anzahl der autonomen Systemen bis zum Ziel (Zusatzinformation IGP-Metrik: wie lange dauert es durch ein autonomes System)

- Distanzvektorprotokolle

Diese Protokolle speichern nur die Distanz zum Ziel

Beispiel: RIP (Routing Information Protocol), Metrik: Anzahl der Hops

EIGRP (Enhanced Interior Gateway Routing Protocol), Metrik: Bandbreite, Auslastung, Delay, Zuverlässigkeit

EIGRP kennt die ganze Topologie im System, speichert diese aber nicht direkt ab.

- Link-State-Protokolle

Diese Protokolle kennen die ganze Topologie im System. Daraus berechnet sich jeder Router die besten Routen zu allen Zielen.

Beispiel: OSPF (Open Shortest Path First), Metrik: Bandbreite

5) Algorithmen zur Bestimmung des kürzesten Weges

- Bellman Ford (RIP)
- Dijkstra (OSPF)
- DUAL (EIGRP)

Dijkstra Algorithmus

Ablauf:

1. Startknoten mit 0 markieren, alle anderen mit ∞ : 'Distanz' und 'besucht' merken 2. Solange es unbesuchte Knoten gibt:

- Jenen Knoten mit der kürzesten Distanz wählen
- Als besucht markieren
- Für alle unbesuchten Knoten die Distanz berechnen
- Falls der Wert kleiner ist, als der aktuelle, diese speichern

6 Aufbau des Internets

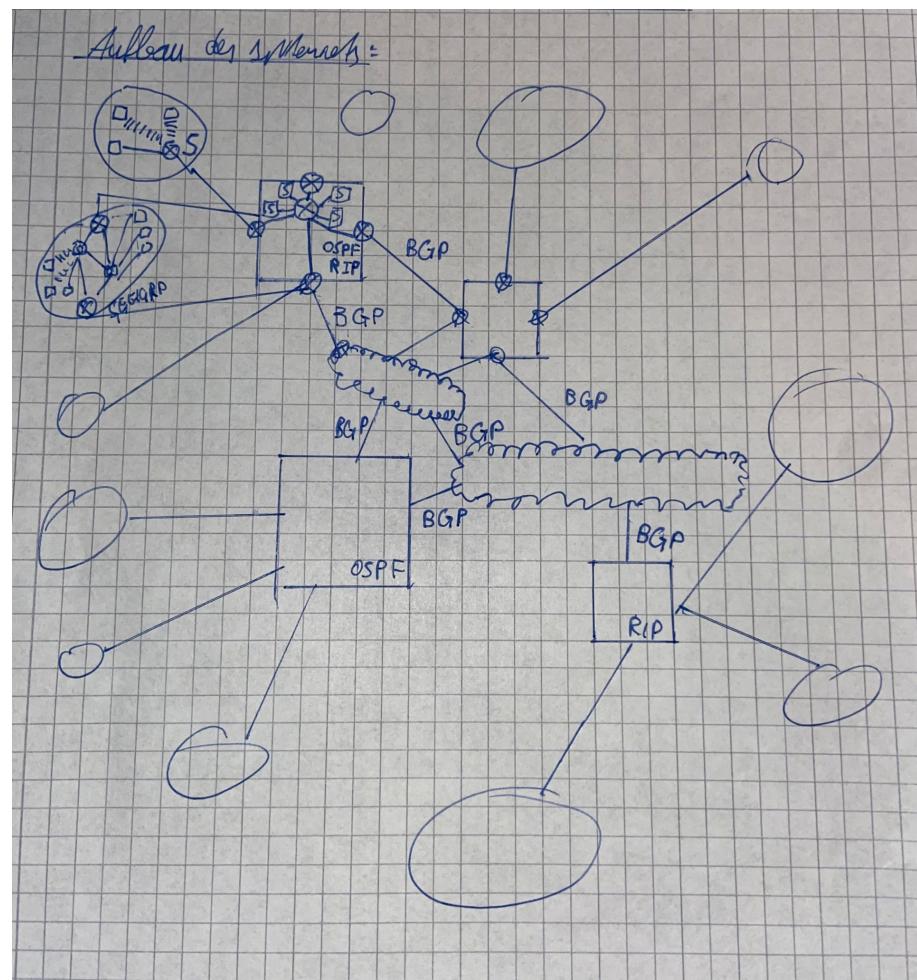


Abbildung 6.1: Aufbau des Internets

7 IPv6

Eine IPv6-Adresse ist eine 128 Bit Zahl. Es gibt 2^{128} IPv6-Adressen ($340 \cdot 10^{36}$).

Schreibweise einer IPv6-Adresse

- Hexadezimale Schreibweise (32 Zeichen)
- Gruppen von 16 Bit mit : getrennt
- Führende Nullen werden in jeder Gruppe weggelassen
- Einmalig kann der längste Block an Nullen mit :: ersetzt werden

Bsp:

2001:ABAD:0000:0430:0000:0000:00C9:0001

2001:ABAD:0:430:0:0:C9:1

2001:ABAD:0:430::C9:1

Subnetzmaske

- trennt in Netz- und Hostteil
- nur noch Präfix-Notation
- Es wird fast nur /64 verwendet

Idee von IPv6

- mehr IP-Adressen
- Problem: alle Protokolle die IPv4 verwenden müssen erneuert werden
- Alte Fehler/Security-Probleme beheben
- leichterer Header

Übergang von IPv4 zu IPv6

- Dualer Stack
- Translation

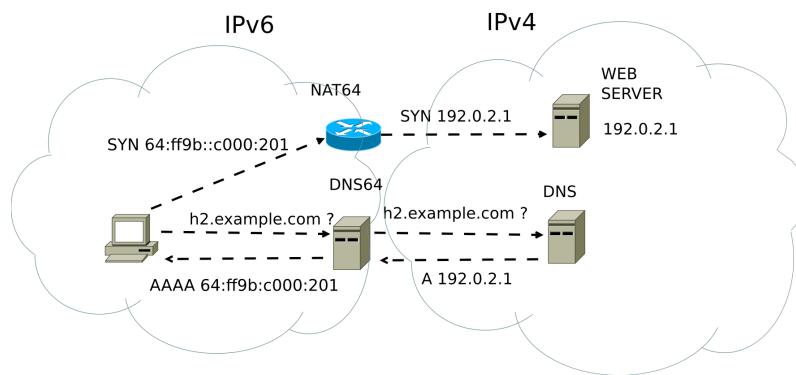


Abbildung 7.1: IPv4-IPv6 Translation mit NAT64

- Tunneling

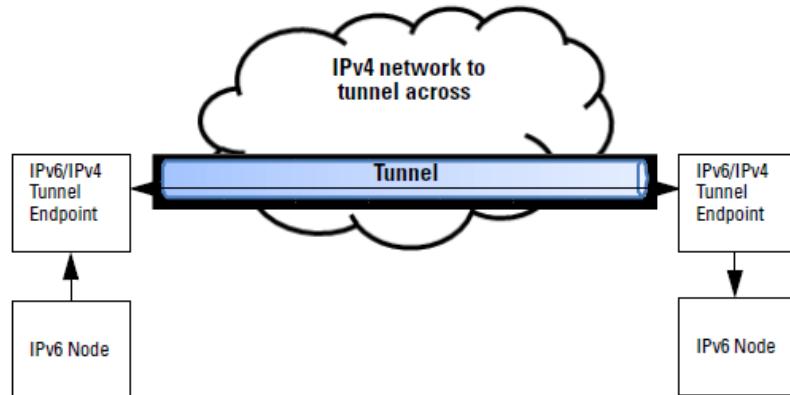


Abbildung 7.2: IPv4-IPv6 Tunneling

Kommunikationsarten

- Unicast
- Multicast
 - ff02::1 ... all-nodes-multicast (Broadcast)
 - ff02::2 ... all-router-multicast
- Anycast (der 'nägeste' einer Gruppe bekommt den Anycast)

IPv6-Unicast Adressen

- Global Unicast Adressen 2000-3fff (vgl. öffentliche IP)
- Link Local Adressen fe80-febf (für das lokale Netz, nicht routbar)

- loopback ::1 (vgl. IPv4 127.0.0.1)
- Unspecified Adress :: (vgl. IPv4 0.0.0.0)
- Unique Local fc00-fdff (vgl. IPv4 NAT)
- Embedded IPv4

IP-Konfiguration

- statisch (GUA, LLA)
- dynamisch
 - SLAAC
 - SLAAC mit stateless DHCPv6 Server
 - DHCPv6

SLAAC

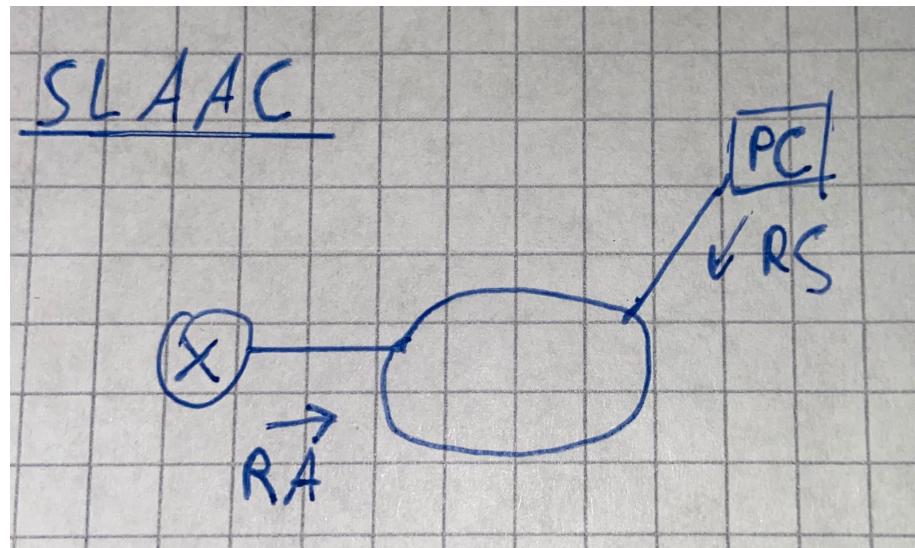


Abbildung 7.3: SLAAC

Router senden (ca. alle 200s) ein RA-Paket (Router Advertisement) aus. Dies enthält die wichtigsten Informationen für die Hosts (Präfix, Präfix-Länge, Default-Gateway). Die Hosts geben sich dann selbst die IPv6-Adresse.

Hostteil

- Zufallszahl (ND-Protokoll)
- EUI-64 (MAC-Adresse)

Die Hosts können RS (Router Solicitation) Pakete aussenden um das RA-Paket anzu fordern.

8 WLAN (Wireless Local Area Network)

Bei einem drahtlosen Netzwerk findet die Übertragung ohne Kabel statt. Es werden elektromagnetische Wellen über die Luft übertragen. Für die drahtlose Übertragung im Netzwerk bedeutet dies, dass sich Layer 1 und Layer 2 ändern. Die darüber liegenden Layer bleiben unverändert. Durch diese Änderung der Übertragungsart ergeben sich einige Vorteile aber auch Nachteile.

| Vorteile | Nachteile |
|---|--|
| <ul style="list-style-type: none"> + BYOD: bring your own device + Kosten: Besonders in bestehenden Gebäuden + Anpassungsfähigkeit | <ul style="list-style-type: none"> - Geteiltes Medium für viele Teilnehmen - Störungen - Geschwindigkeit und Reichweite - Security |

Antennen

Antennen sind die Grundlage für eine Übertragung über die Luft. Sie geben ein Signal in die Luft ab (Senderantenne) und können es auch wieder aus der Luft aufgreifen (Empfängerantenne). Je nach Anwendung eignen sich verschiedene Arten von Antennen.

- Omnidirektionale Antennen: senden in alle Richtungen (Kugel)
- Direkte Antennen: Können gezielt senden
- MIMO (multiple input multiple output) Antennen: aktuell meist 8 Antennen

Arten von Wireless Netzwerken

Wie auch schon bei den verkabelten Netzen unterscheidet man Netze nach ihrer Größe. Je nach Größe ergeben sich unterschiedliche Anforderungen und Schwierigkeiten.

- WPAN: kurze Distanz (ca 10m), Frequenz meist 2.4 GHz z.B. Bluetooth, Zigbee
- WLAN: mittlere Distanz (ca 100m), Frequenz ist 2.4 GHz oder 5 GHz z.B. Wifi
- WMAN: große Distanz (kann sehr unterschiedlich sein), Frequenz zwischen 2 und 66 GHz z.B. Wifi, WiMax

- WWAN: riesige Distanzen (bis zu 50km), Frequenz zwischen 2 und 66 GHz z.B. WiMax

Technologien von Wireless Netzwerken

Es gibt unterschiedliche Technologien die drahtlos übertragen. Je nach Reichweite und Anwendungsgebiete sind unterschiedliche Technologien sinnvoll. Nicht alle Technologien sind dazu geeignet oder dafür entworfen um Netzwerksdaten zu übertragen. Manche Technologien können dies trotzdem umsetzen.

- Wifi (IEEE 802.11)
 - Bluetooth (IEEE 802.15)
 - WiMax (IEEE 802.16)
 - Satelliten Breitband: kann als Internetzugang genutzt werden, z.B. Starlink (Oktober 2023 ca. 5000 Geräte, in einer Entfernung von 500 bis 600km, beantragt sind 22.000 Satelliten)
 - Mobilfunk Breitband: viele verschiedene Standards die meist nach gravierenden Änderungen (Generationen) unterteilt werden. Bei der Änderung in eine neue Generation ist die Geschwindigkeit immer ein entscheidender Faktor (3G → 10x → 4G → 100x → 5G).

8.1 Wifi (802.11)

Elektromagnetische Welle

Gesendet wird mit elektromagnetischen Wellen. Diese kennt man vom sichtbaren Licht. Dort nimmt der Mensch unterschiedliche Wellenlnge als verschiedene Farben wahr. Jene Wellenlnge die zum bertragen von Wifi genutzt werden liegen auerhalb des sichtbaren Lichts. Dest lnger die Welle ist, desto krzer ist seine Frequenz (indirekt Proportional). Kurze Wellen besitzen mehr Energie als lange Wellen.

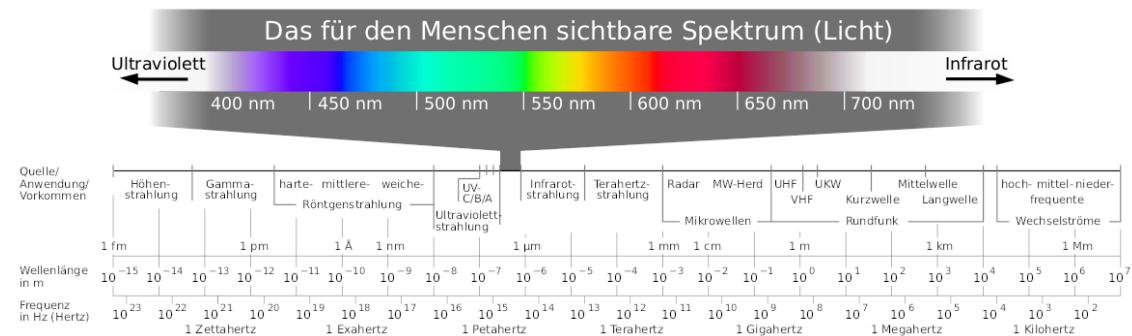


Abbildung 8.1: Elektromagnetisches Spektrum

- 2.4 GHz (1-10dm) UHF: Ultra High Frequency
- 5 GHz (1-10cm) SHF: Super High Frequency

Für das 5G Netz wurden neue Frequenzbereiche festgelegt und versteigert.

- Mobilfunk: 600 MHz bis 6 GHz
- WLAN: 24 GHz bis 40 GHz

Komponenten im WLAN

- Endgeräte: mit Netzwerkkarten und Antennen
- Wireless Router: Multifunktionsgeräte mit eingebautem Switch, Router, Modem Access Point,...
- Access Points: Schnittstelle zwischen dem drahtlosen Netz und dem verkabelten Netz. Man unterscheidet zwischen Autonomen-Access-Points (schwer erweiterbar) und Controller-Based-Access-Points.

Wifi Frame

| | | |
|------------------|--|---------|
| Frame Control | Metainformationen z.B. Protokoll, Art des Frames,... | 2 Bytes |
| Duration | Übertragungsdauer, aufgrund unterschiedlicher Framelänge | 2 Bytes |
| Address 1 | Empfänger MAC-Adresse | 6 Bytes |
| Address 2 | Sender MAC-Adresse | 6 Bytes |
| Address 3 | MAC BSSID (WLAN Segment) | 6 Bytes |
| Sequence Control | Hängt vom AP ab | |
| Address 4 | MAC-Adresse vom Access Point | 6 Bytes |
| Frame Body: | Header der restlichen Layer und Daten | |
| FC | Fehlerüberprüfung mit CRC | 4 Bytes |

Operations-Modi

- Ad Hoc: Peer-to-Peer Netzwerk ohne Router
- Infrastruktur: Dahinter ein verkabeltes Netz
- Tethering: Hotspot zur Weiterleitung zwischen zwei Netzen

Kollisionen (CSMA/CA)

Wireless Netzwerke nutzen eine Half Duplex Medium zum Senden. Man kann zeitgleich senden und empfangen. Zusätzlich ist es ein Shared Medium, das heißt viele Teilnehmer sind mit dem gleichen Medium verbunden. Somit kann es zu Kollisionen kommen (CSMA - Carrier Sense Multiple Access). Wifi löst das Problem mit Collision Avoidance (CA), es versucht also Kollisionen zu vermeiden. Falls gerade keiner sendet wird um Zeit beim Access Point angefragt. Dann erhält man einen Zeitslot indem man seine Daten senden und empfangen kann.

Verbinden mit einem Accesspoint

- AP finden (aktiv, passiv)
- Authentifizieren: SSID, Passwort, Network Mode (a, b, g,...), Security (WPA, WPA2,...), Channel
- Verbindung herstellen

Channels

Die Frequenzen werden in kleinere Bereiche aufgesplittet. Gleiche Channels können sich gegenseitig stören. Überlappende Access Points sollten verschiedene Channels nutzen.

- 2.4 GHz: Europa 13 Channels (1, 6 & 11 nicht überlappend)
- 5 GHz: 24 Channels (alle ohne Überlappung)

WLAN-Angriffe

- Datendiebstahl: Shared medium → Verschlüsselung
- DoS: falsch konfiguriert, Störsender,...
- Rogue Access Point: zusätzlichen falschen AP ins Netz eingefügt
- Evil Twin: einen AP einfügen, der gleich aussieht aber in ein anderes Netz führt

Sicherheit und Verschlüsselung

- SSID Beacon verbergen (passic)
- MAC-Adressen filtern (L2 Security)
- Authentifizierung
 - Open: ohne Password (nicht empfohlen)
 - Shared Key: WEP, WPA (TKIP+AES), WPA2, WPA3

Bei WPA2 unterscheidet zwei Varianten zum Authentifizieren:

- Personal: ein Passwort für alles (PSK, Pre Shared Key), eher im privaten Bereich
- Enterprise: Anmeldung mit Username und Passwort, man meldet sich bei einem Server (z.B. RADIUS), eher im Firmenbereich

WPA2-Personal Handshake: Pre-Shared-Key (4-Way)

Zum Austausch der Schlüssel zwischen dem Access Point und dem Client findet ein 4-Way-Handshake statt. Dabei werden die benötigten Schlüssel generiert. Zum Generieren der Schlüssel muss das Passwort nie übertragen werden, deshalb nennt man die Variante auch PSK (Pre-Shared-Key). Der Schlüssel wurde also davor schon ausgemacht.

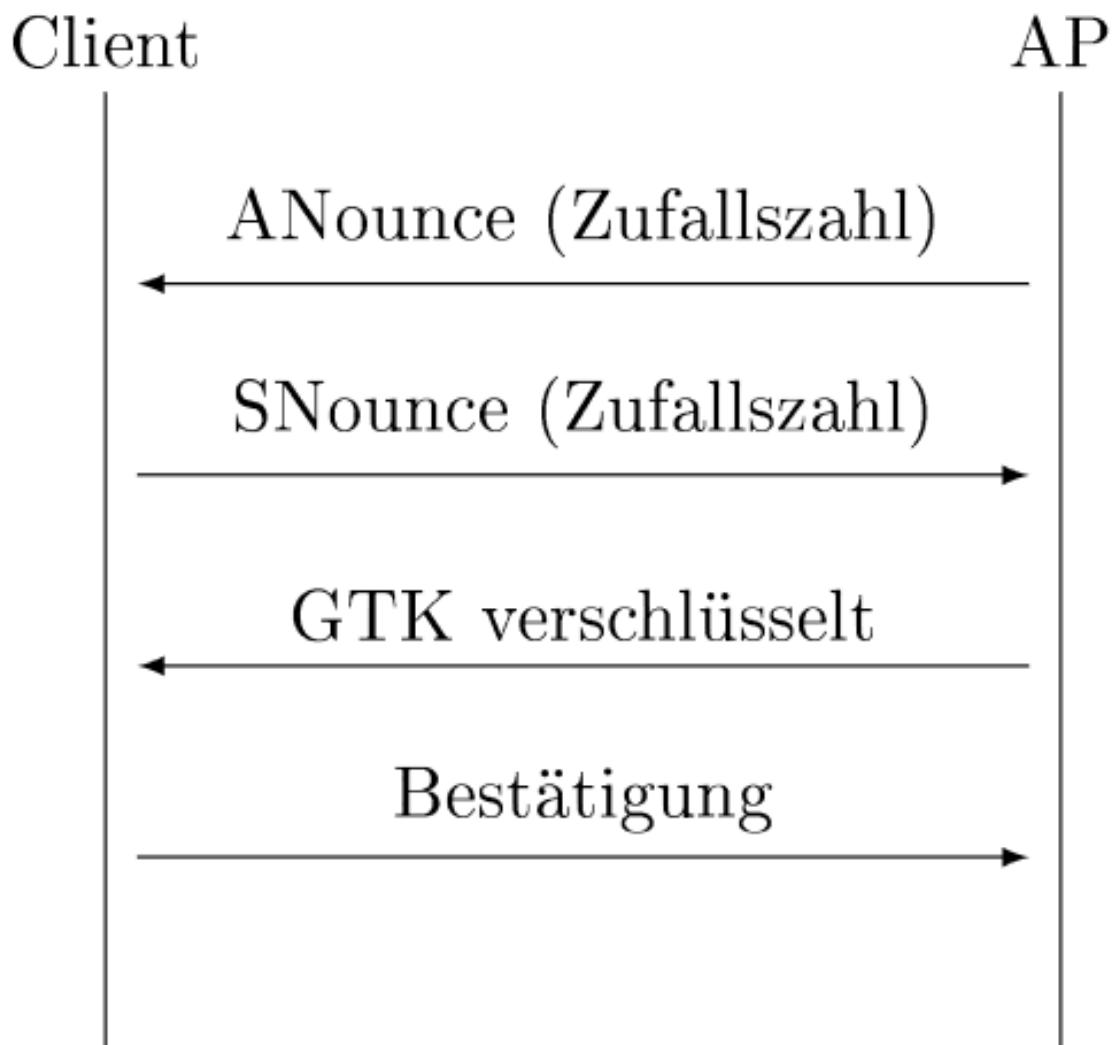


Abbildung 8.2: WPA2 Personal Handshake

PTK (Pairwise Transient Key): für Unicasts, jeder hat seinen eigenen Schlüssel mit dem AP

$\text{PTK} = \text{PRF}(\text{Pwd} + \text{ANounce} + \text{SNounce} + \text{APMAC} + \text{ClientMAC})$

PRF (Pseudo Random Function): ist eine Pseudo-Zufallsfunktion die dann den Schlüssel erzeugt und den Geräten bekannt ist.

GTK (Group Temporal Key): für Broadcast und Multicasts im Netz, für alle Teilnehmer gleich.

Das Passwort wird nie über das Medium ausgetauscht, deshalb nennt man das Verfahren Pre Shared Key. Die Nachricht wird nach Layer 2 verschlüsselt. Dieser kann nicht

verschlüsselt werden, da der Access Point die Frames identifizieren muss. Danach im verkabelten Netz, wie sonst auch immer, wird wieder nach Layer 4 verschlüsselt (z.B. mit TLS).

9 Network Security

Netzwerkangriffe können auf unterschiedliche Arten, unterschiedlichen Ebenen und verschiedene Protokolle stattfinden. Deshalb muss bei einem Security-Konzept möglichst alles berücksichtigt werden.

| Angriffe | OSI-Modell | Abwehr |
|--|---|--|
| Social Engineering Passwörter Pretexting | L8-Mensch | Schulungen Vernünftig und vorsichtiges handeln |
| SQL-Injection Wurm, Virus, Trojaner Ransomware, Spyware Protokolle (HTTP, FTP, Telnet, DHCP, DNS,...) | L7-Application L6-Presentation L5-Session | Firewall, IPS, IDS, ESA, WSA Eingabeüberprüfung gute Software & Protokolle End-Point-Detection Anti-Virus, Updates |
| DDOS → TCP (SYN-Flood, → UDP | L4-Transport | IPS, IDS Firewall, ACL |
| Routing, DDoS MITM, IP-Spoofing Protokolle (ICMP, | L3-Network | Firewall, ACL, IPS, IDS sichere Protokolle (IPsec) |
| MITM: ARP-Spoofing, MAC-Spoofing DDOS: MAC-Flooding Protokolle (STP, CDP) | L2-Data Link | MAC-Filter, AAA sichere Protokolle Verschlüsselung |
| DoS (Störsender, Zerstörung der Infrastruktur) Physischer Zugang MITM, Hardware | L1-Physical | Zutrittskontrolle Backups |

Dem Angreifer reicht eventuell ein einziger Angriffspunkt im Netz. Meist sind die User (Personen) das größte Problem. → "Der Angreifer muss nur einmal gewinnen"

Sicherheitsrichtlinien

| User | Unternehmen |
|---|--|
| <ul style="list-style-type: none"> • Passwörter (Mindestlänge, eins pro Account, keine persönlichen Daten) → Passwortmanager, 2FA • Datenverwaltung (Wann?, Wo?, Welche?, Wann?,...) • Firewall • Updates (OS, Software) • Antivirus/Antispysoftware • Vernünftig handeln | <ul style="list-style-type: none"> • Passwortrichtlinien, User Verwaltung, Recht vergeben • Firmengeräte, spezielle Rechte, wie beim User • DMZ, VPN, Firewall, IPS, IDS, WSA, ESA • Zugangskontrollen • Schulung der Mitarbeiter • Backups • Pen-Testing • Risikoanalyse → Schwachstellen kennen • Verhaltensanalyse • Datatransfer sichtbar machen |

9.1 Firewall

Mit einer Firewall kann der eingehende/ausgehende Datenverkehr kontrolliert, protokolliert und gefiltert werden (sperren, freigeben).

Unterscheidung nach Position

- **Personal Firewall** (am eigenen Gerät) z.B. Windows Defender, UFW,...
- **External Firewall** (zwischen lokalen & globalen Netz) z.B. ASA, Fortinet, Barracuda, PFSense,...

Unterscheidung nach Funktion

- (L4) **Paketfilter:** IP-Adressen, Ports z.B. ACL
- (L4) **Stateful Inspection:** Untersucht die ganze Sitzung (mehrere Aufrufe zu z.B. gleiche IPs)
- (L7) **Application Firewall:** Proxy Server
- (Daten) **Deep Paket Inspection Firewall**

Eine falsch konfigurierte Firewall bietet keinen Schutz. Eine Firewall muss ständig gewartet und aktualisiert werden.

9.2 IDS & IPS

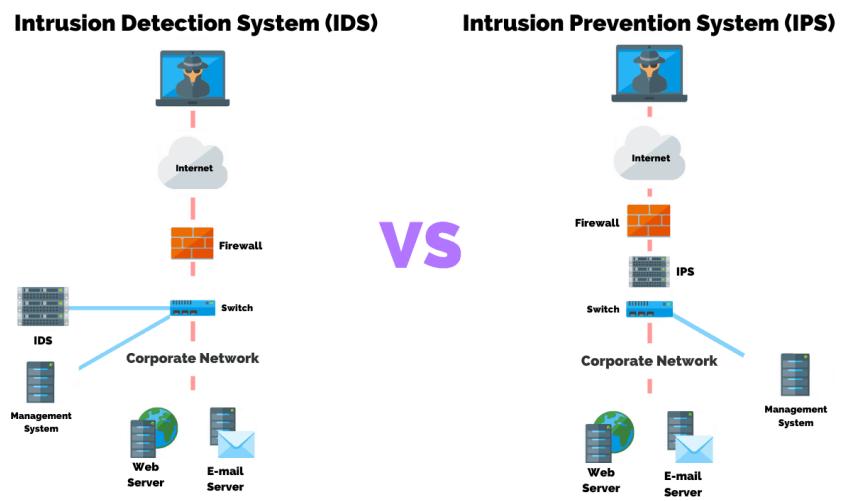


Abbildung 9.1: IPS & IDS

| IDS (Intrusion Detection System) | IPS (Intrusion Prevention System) |
|---|---|
| Wird nur parallel informiert + schneller (da es parallel ist) - nur Warnungen | Alles muss über IPS - langsamer (da seriell) + sicherer |

9.3 Honeypot

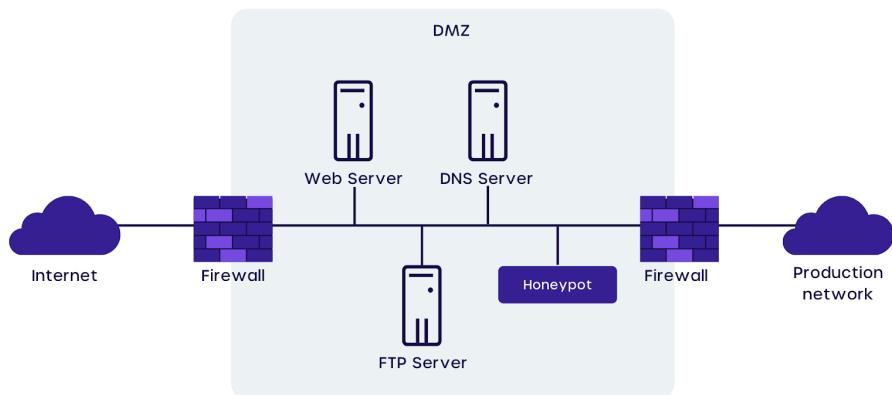


Abbildung 9.2: Honeypot

Bei einem Honeypot werden bewusst veraltete Software & Hardware für Angreifer als Köder aufgestellt.

9.4 VPN

Erstellt eine verschlüsselte Verbindung zu einem entfernten VPN-Server.

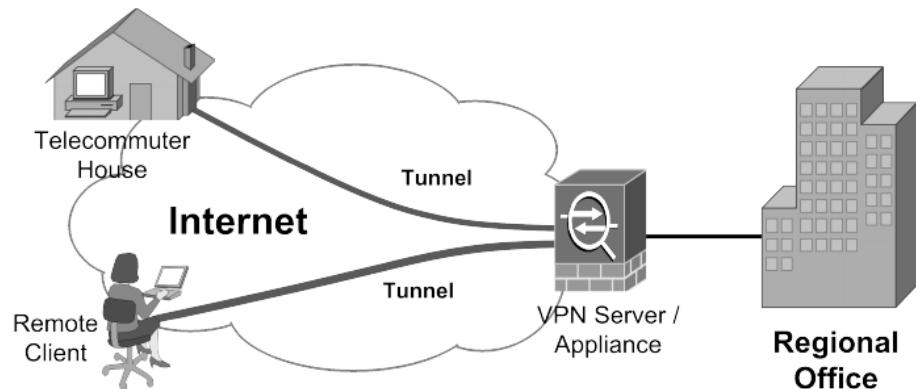


Abbildung 9.3: VPN

Arten von VPN:

- Ende-zu-Ende VPN
- Ende-zu-Netz VPN
- Netz-zu-Netz VPN

9.5 ESA (Email Security Appliance) & WSA (Web Security Appliance)

Funktioniert wie ein Proxy-Server.

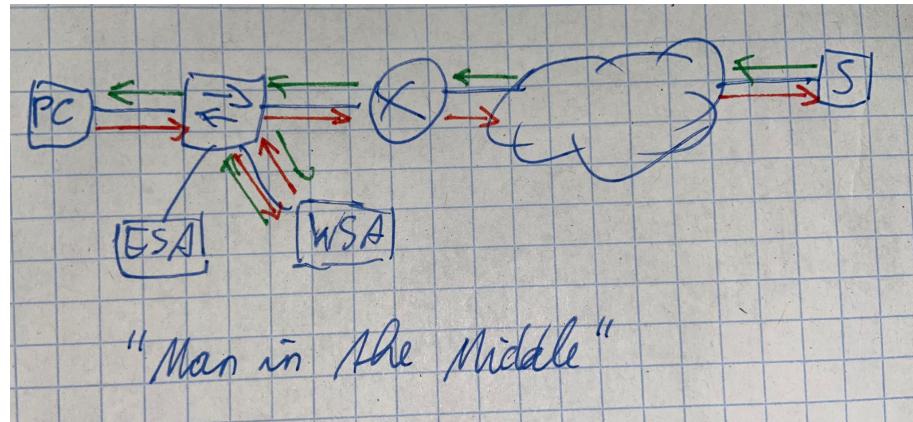


Abbildung 9.4: ESA & WSA

9.6 IPsec

Ziel: Sicheres Protokoll zur Datenübertragung

- **Transport-Modus:** Verschlüsselung ab L4 und fügt eine Authentifizierung in den Header ein.
- **Tunnel-Modus:** Alles wird verschlüsselt und es wird ein neuer verschlüsselter Header angehängt. Darin stehen die wichtigsten Felder (MAC, IP,...) + Authentifizierung

10 Hashfunktionen

Bei einer Hashfunktion ist die Wertemenge meist wesentlich kleiner als die Lösungsmenge.
 Die Elemente der Wertmenge können normalerweise eine beliebige Länge haben.
 Die Elemente der Lösungsmenge haben meist eine fixe Länge.

- Anfangsbuchstabe: Hallo → H
 Tim → T
- Postleitzahl: Grins → 6591
 Neustift → 6167
- CRC ...

Kryptographische Hashfunktionen

Kryptographische Hashfunktionen müssen spezielle Eigenschaften erfüllen. Die wichtigste Eigenschaft ist, dass es sich um eine Einwegfunktion handelt.

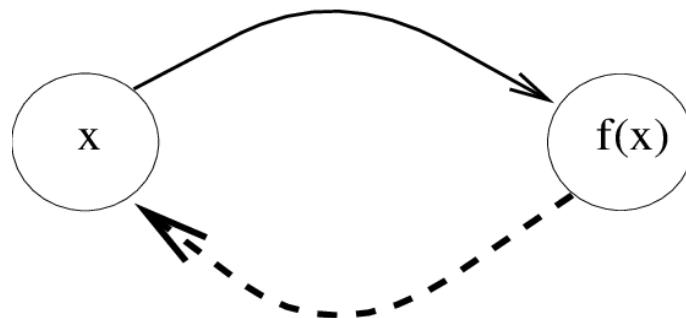


Abbildung 10.1: Einwegfunktion

Eigenschaften

- **Einwegfunktion:** Unumkehrbarkeit muss so gut wie möglich gegeben sein
- Diffusion ('Lawineneffekt'): kleine Änderung in der Eingabe bewirkt die ganze Ausgaben.
- Konfusion: Von Hashwert kann man keine Rückschlüsse auf den Eingabewert machen
- Eindeutigkeit

- Kollisionsresistenz: Die Wahrscheinlichkeit, dass Kollisionen vorkommen soll so klein wie möglich sein (im besten Fall gleich Null sein)

Hash-Algorithmen

- MD5: 128 Bit Hashwert, unsicher!
- SHA (secure hash algorithm)
 - SHA1: 160 Bit Hashwerte unsicher!
 - SHA2
 - SHA224, SHA256, SHA 384, SHA512
 - SHA3: grundlegend anders, 224, 256, 384 Bitwerte → auch frei wählbar
- GOST, Whirlpool

Ablauf SHA-256

- Block erstellen (Auffüllen, Startblock erstellen, Wurzel von Primzahlen)
- Bitrotation, Zeilen integrieren (Diffusion)
- XOR, Bitshift (viele Runden)
- Auswahlfunktion, Mehrheitsfunktion (Einwegfunktion)

Zusatz: Message Authentication Code z.B. HMAC: Schlüssel-Hash-Nachrichtauthentifizierung, Pre-Shared-Key

Angriffe

- Brute-Force
- Phising
- Wörterbuchangriff
- Algorithmus nutzen
- Rainbow-Table (viel Speicher)
 - viele Hashwerte als Kette gespeichert

Passwörter

1. lokale Speicherung
 - Länge des Passworters (mind 12 Zeichen)
 - keine Wörter, persönliche Informationen
 - Buchstaben/Zahlen/Symbole

- Jedes Passwort nur einmal verwenden
- Passwortmanager verwenden oder MFA als Alternative

2. Speicherung am Server

- Klartext ↳ Zugriff auf Datenbank, Admin, MITM
- Gehashed ↳ MITM, gleiche Passwörter erkennbar
- Gehashed + Salt: Mitm, Brute-Force
- Gehashed + Salt + Pepper
 - Salt: Zufällige Zeichenkette die im Klartext in der Datenbank gespeichert wird → Jeder bekommt eigenen Hashwert
 - Pepper: Zufällige Zeichenkette die NICHT in der Datenbank steht

3. Austausch Client-Server

- PAP unsicher!

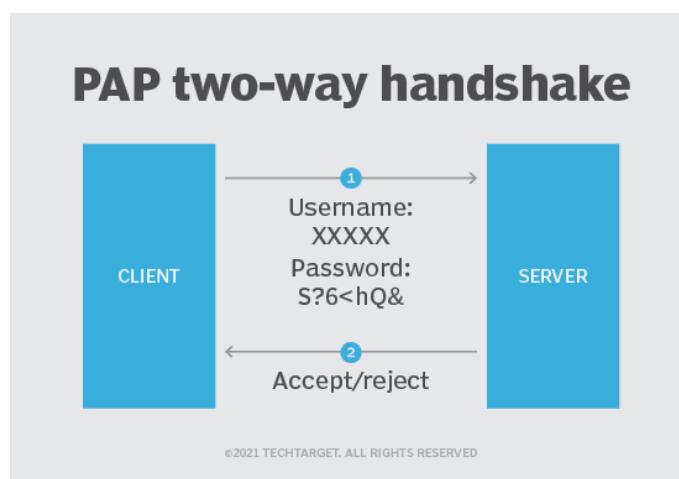


Abbildung 10.2: PAP

- CHAP: Bei CHAP wird bei jedem Anmeldeversuch eine Challenge (Zufallszahl) gesendet. Der Client hashet sein Passwort mit der Zufallszahl und sendet es dem Server. Der Server kann dann das Passwort in der Datenbank mit der gesendeten Zufallszahl hashen und es mit dem Client-Hash vergleichen. Somit wird zum einen, das Passwort nie im Klartext gesendet und zum anderen kann der gesendete Hash nicht noch einmal gesendet werden, da die Zufallszahl 'einzigartig' ist und bei jedem Anmeldeversuch anders ist.
→ MITM Anmeldung mit dem Hashwert ist durch die Challenge nicht mehr möglich



Abbildung 10.3: PAP

Alternativen: MS-CHAPv1, MSCHAPv2, EAP, PEAP,...

AAA: Autorisierung (Was?), Authentifizierung (Wer?) & Accounting (Wann?)

Protokolle: RADIUS, TACACS+

PKI (Public Key Infrastruktur)

Digitale Zertifikate sind für die Authentifizierung eines öffentlichen Schlüssels und seiner zulässigen Anwendung bzw. Geltungsbereich

Vergleich:
Zertifikat → Pass
digitale Signatur → Foto

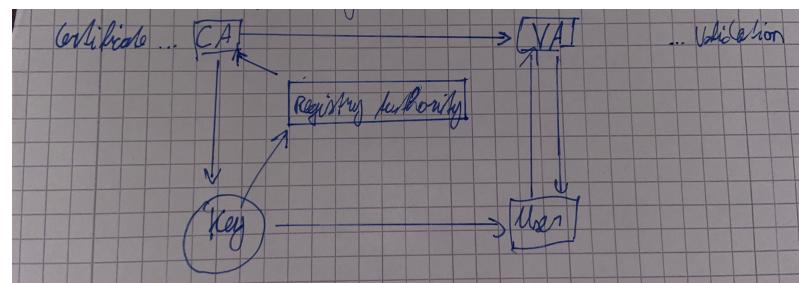


Abbildung 10.4: Public Key Infrastruktur

Alternative: Web of Trust



| Zusammenfassung | typische Protokolle |
|-------------------------------|---------------------|
| AES/DES | HTTPS |
| RSA | IPsec |
| Diffie-Hellman | VPN |
| Hashfunktionen (SHA256,...) | SSH |
| MAC (bzw. HMAC) | PSK (WLAN) |
| Authentifizierung (AAA, CHAP) | RADIUS |
| PKI | ... |

HTTPS: HTTP + TLS (SSL)

TLS ... transport layer security, Verschlüsselung nach Layer 4

- Symmetrische Verschlüsselung
- Asymmetrischer Schlüsselaustausch (ab TLS 1.3 Diffie-Hellman)
- Authentifizierung (Server), PKI, RSA
- Hashfunktionen (SHA256,...)
- Sicherung der Nachrichtenintegrität (HMAC)

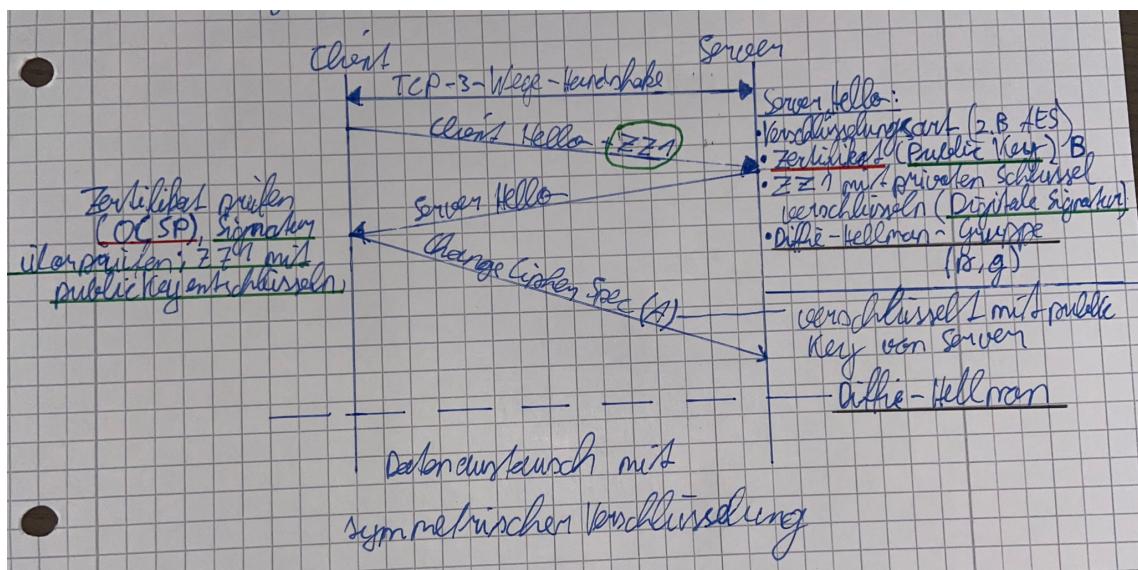


Abbildung 10.5: HTTPS Verbindungsaufbau

Abbildungsverzeichnis

| | | |
|------|---|----|
| 1.1 | Bustopologiearten | 6 |
| 1.2 | CAN-Bus Aufbau | 9 |
| 1.3 | I2C-Bus Bitsetzung | 11 |
| 1.4 | I2C-Bus Steuersignale | 11 |
| 1.5 | I2C Ablauf | 12 |
| 2.1 | Finite State Machine: Ampel | 13 |
| 4.1 | OSI-Modell Datenübertragung | 20 |
| 4.2 | Cisco CLI | 21 |
| 4.3 | Glasfaserkabelarten | 23 |
| 4.4 | Baum- und Stern topologie | 24 |
| 4.5 | Ethernet Frame | 25 |
| 4.6 | Layer 2 & 3 Adressierung | 26 |
| 4.7 | ARP-Spoofing | 27 |
| 4.8 | CIDR Beispiel | 30 |
| 4.9 | VLSM Beispiel | 31 |
| 4.10 | L4-Adressierung | 33 |
| 4.11 | TCP 3-Way-Handshake | 34 |
| 4.12 | TCP 2-Way-Handshake | 34 |
| 4.13 | Layer 4 Segmentierung | 35 |
| 4.14 | Request/Response Modell | 36 |
| 4.15 | DNS-Hierarchie | 37 |
| 4.16 | DHCP-Handshake | 37 |
| 4.17 | Request/Response Modell | 38 |
| 4.18 | Position der Wildcardmask | 41 |
| 4.19 | Static NAT, 1:1 Mapping | 41 |
| 4.20 | NAT mit PAT, n:1 Mapping | 41 |
| 6.1 | Aufbau des Internets | 47 |
| 7.1 | IPv4-IPv6 Translation mit NAT64 | 49 |
| 7.2 | IPv4-IPv6 Tunneling | 49 |
| 7.3 | SLAAC | 50 |



| | | |
|------|--|----|
| 8.1 | Elektromagnetisches Spektrum | 53 |
| 8.2 | WPA2 Personal Handshake | 56 |
| 9.1 | IPS & IDS | 60 |
| 9.2 | Honeypot | 60 |
| 9.3 | VPN | 61 |
| 9.4 | ESA & WSA | 62 |
| 10.1 | Einwegfunktion | 63 |
| 10.2 | PAP | 65 |
| 10.3 | PAP | 66 |
| 10.4 | Public Key Infrastruktur | 66 |
| 10.5 | HTTPS Verbindungsaufbau | 67 |

I Quellcodeverzeichnis