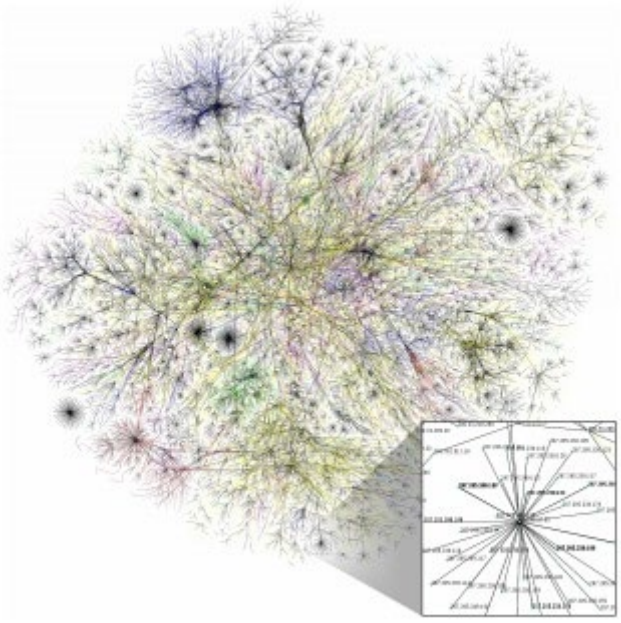
Ostfalia
Hochschule für angewandte Wissenschaften

Labor für Kommunikationssysteme

Leitung: Prof. Dr.-Ing. María Dolores Pérez Guirao



Versuch:

Routing in Netzwerken

Wintersemester 2022

Gruppe: _____

Datum: _____

Teilnehmer:

Name: _____ Matr.-Nr.: _____

Name: _____ Matr.-Nr.: _____

Name: _____ Matr.-Nr.: _____

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Grundlagenwissen

1.1 Rechnerkommunikation

Unter dem Begriff Kommunikation wird allgemein die Übertragung einer Nachricht von einem Sender zu einem Empfänger verstanden. Bei jeder Kommunikation müssen bestimmte Voraussetzungen, wie zum Beispiel die Festlegung der Kommunikationsart und des Übertragungskanals sowie die Aufstellung von Regeln für die Zeichenumsetzung, erfüllt sein.

1.1.1 Kommunikationsmodelle

In Modellen für die Kommunikation offener Systeme wird das komplexe Problem der Datenkommunikation innerhalb von Netzwerken in kleinere Teilprobleme (Ebenen, Schichten, Layer) gegliedert. Für jede Schicht existiert eine Beschreibung des Funktionsumfanges sowie des Verhaltens an den Schnittstellen zu den benachbarten Schichten. Dabei wird nur festgelegt, welche Funktionen von den einzelnen Schichten zu erfüllen sind. Es wird nicht festgelegt, wie diese Schichten zu implementieren sind. Einzelne Schichten können weiter unterteilt werden oder ungenutzt bleiben, wenn es bei einem konkreten Netzkonzept zweckmäßig erscheint. Viele Hersteller haben in der Vergangenheit eigene Standards für die Kommunikation zwischen ihren Geräten entwickelt. Die ausschließliche Verwendung von Hard- und Software eines einzigen Herstellers ist sicher nur in ganz wenigen Ausnahmefällen sinnvoll. Es war daher notwendig, herstellerunabhängige Richtlinien zu entwickeln, die eine Kommunikation zwischen heterogenen Systemen ermöglichen. Zwei inzwischen fest etablierte Standards sind das *OSI-Modell* und das *Projekt 802-Modell*. Beide beschreiben den komplexen Sendevorgang in einem Netzwerk und sind zueinander kompatibel.

1.1.1.1 OSI-Modell

Das OSI- Referenzmodell (Open Systems Interconnection) der International Standard Organisation ist eines der derzeit wichtigsten theoretischen Modelle zur Beschreibung der Kommunikation über Datennetze. Alle für die Kommunikation notwendigen Funktionen werden hierarchisch so gegliedert, dass jede Schicht (Layer) als eigenständige Einheit angesehen werden kann, die einen Dienst für die nächst höhere Schicht erbringt und ihrerseits Dienste von den ihr untergeordneten Schichten in Anspruch nimmt. Eine direkte Kommunikation zwischen Protokollen derselben (höheren) Schicht zweier Datenendeinrichtungen ist nicht möglich. Die Daten müssen erst alle untergeordneten Protokollebenen beim Absender durchlaufen, bevor sie in der untersten Schicht (Bitübertragungsschicht, physikalische Schicht, Physical Layer) über ein geeignetes Medium übertragen werden können. Beim Empfänger findet dieser Prozess in umgekehrter Richtung statt (Abbildung 1).

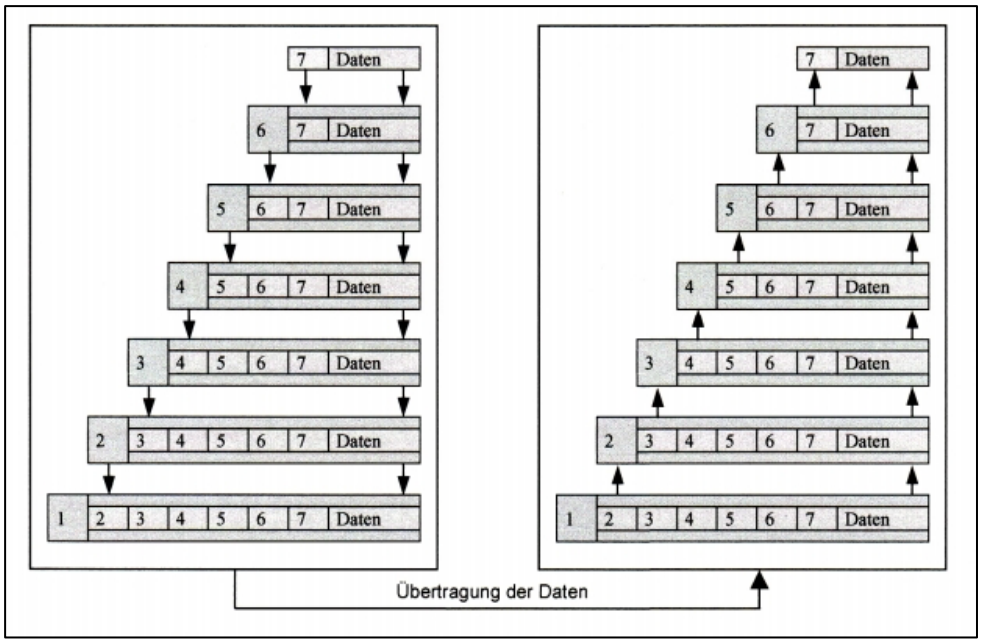


Abbildung 1: Weg der Daten durch die Protokollebenen

Tabelle 1 Stellt die Aufgaben der einzelnen OSI-Schichten dar.

Tabelle 1: Funktionen der OSI-Schicht

	Bezeichnung	Hauptaufgaben
7	Anwendungsschicht (Application Layer)	Identifikation der Kommunikationsparameter, Beurteilung der Verfügbarkeit der Kommunikationspartner, Berechtigungsprüfung, Wahl der Übertragungsparameter (Dienstqualität, Priorität, etc.)
6	Darstellungsschicht (Presentation-Layer)	Code- und Alphabetwandlung, Formatanpassung, Wahl der geeigneten Syntax (entsprechend der Anwendung)
5	Sitzungsschicht (Session-Layer)	Aufbau und Aufrechterhaltung logischer Verbindungen, Verbindungsidentifikation, Dialogsteuerung
4	Transportschicht (Transport-Layer)	Aufbau und Überwachung von Duplex-Übermittlungspfaden, Anpassung an unterschiedliche Netzeigenschaften, Ende-zu-Ende-Fehlererkennung, Segmentierung und Blockbildung, Adressübersetzungen
3	Vermittlungsschicht (Network-Layer)	Verbindungslenkung, Aufbau und Überwachung von Netzverbindungen, Verbindungsmultiplexierung, Netzabhängige Fehlerüberwachung, Flusssteuerung, Verwaltung von Netzressourcen
2	Sicherungsschicht (Link-Layer)	Leitungsaktivierung, Leitungsdeaktivierung, Übertragungssteuerung, Übertragungsfehlerüberwachung, Blocksynchronisation, Wahl des geeignetsten Übertragungspfades
1	Bitübertragungsschicht (Physical-Layer)	Parallel/Seriell-Wandlung, Anpassung an die Physik der unterschiedlichen Übertragungsmedien, Synchronisation von Informationselementen (Bits), Zusammenschaltung von Abschnitten unterschiedlicher Übertragungsmedien, Zustandsüberwachung und Signalisierung

1.1.1.2 Projekt 802 Modell

Das Projekt 802 wurde von der IEEE (Institution of Electrical and Electronic Engineers, US-Normungsinstitut) ins Leben gerufen, um eine Standardisierung für die immer mehr an Bedeutung gewinnenden lokalen Netzwerke zu erreichen. Die speziellen Eigenschaften von LANs (z.B. dezentrale Vermittlung) sollten in einer neuen, vom OSI-Modell abgeleiteten LAN-Architektur berücksichtigt werden. Die Projektbezeichnung ergab sich aus dem Jahr (1980) und Monat (Februar) des Projektbeginns. Dieses Projekt enthält weitere Standards für die physischen Komponenten eines Netzwerks (Schnittstellenkarte und Verkabelung). Die Standards betreffen die beiden untersten Schichten des OSI-Modells. Die Schicht 2 (Sicherungsschicht) des OSI-Modells wird beim 802-Standard in zwei Teilschichten unterteilt (Abbildung 3).

- Die LLC-Teilschicht (Logical Link Control) ist für die Flusskontrolle zuständig. Sie verwaltet die Datenverbindung und definiert logische Schnittstellenpunkte (Service Access Points, SAPs), die für den Transport von Informationen aus der LLC-Teilschicht zu den höheren OSI-Schichten genutzt werden.

- Die MAC-Teilschicht (Media Access Control) enthält Angaben über Zugriffssteuerung und Fehlerbehandlung:

    - Sie steuert den Zugriff der Netzwerkkarten auf die Bitübertragungsschicht, d.h. sie kommuniziert direkt mit der Netzwerkkarte.

    - Sie ist für die fehlerfreie Datenübertragung zwischen den einzelnen Computern verantwortlich. Dies beinhaltet das Herstellen, Aufrechterhalten und Beenden einer Verbindung.

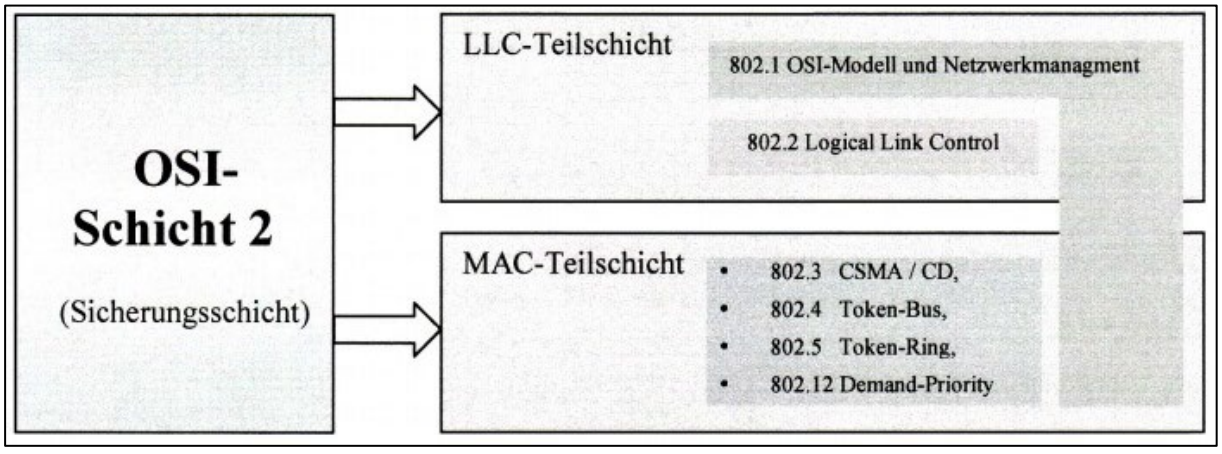


Abbildung 2: LLC- und MAC-Standards des Projektes 802 (Microsoft Press)

1.1.1.3 Kommunikationsprotokolle

Im Gegensatz zu theoretischen Kommunikations- Modellen beinhaltet ein Kommunikations-Protokoll (Schicht-Protokoll, Layer Protocol) präzise Spezifikationen der Vorschriften und Regeln zum Informationsaustausch zwischen mindestens zwei Partnern derselben Schicht (Layer) eines Kommunikationssystems. Dabei werden Regeln für Syntax (z.B. Datenformate), Semantik (Bedeutung zugelassener Kommandos) und Zeitvorgaben („Time out“) festgelegt. Im Rahmen dieses Laborversuches werden nur einige konkrete Protokolle aus der TCP/IP-Protokollfamilie angesprochen, die insbesondere zum Verständnis der Vorgänge auf der OSI-Schicht 3 beitragen.

1.1.2 Vermittlungsprinzipien

Die Aufgabe von Vermittlungssystemen ist das Herstellen von Verbindungen bzw. die Vermittlung von Nachrichten zwischen zwei oder mehr Teilnehmern. Dabei werden Systeme mit zentraler Vermittlung von Systemen mit dezentraler Vermittlung unterschieden. Bei Systemen mit zentraler Vermittlung (z. B. ISDN) kommunizieren zwei Partner nicht direkt miteinander, sondern immer über eine zentrale, übergeordnete Instanz. So geht z. B. jedes Telefongespräch über eine Vermittlungsstelle. Dagegen ist in dezentral vermittelten Systemen ein direkter Datenaustausch mit Teilnehmern im gleichen Segment möglich (z. B. LAN).

1.1.2.1 Paketvermittlung (Packet-Switching)

Der Datenaustausch beruht auf einem relativ einfachen Prinzip: Jede Information bzw. Nachricht wird in einzelne Pakete aufgeteilt. Diese Pakete werden zu ihrem Bestimmungsort geschickt und dort wieder zur ursprünglichen Nachricht zusammengesetzt. Die Leitungswege werden nach der momentanen Netzlast gewählt. Die Verarbeitungsleistung der Vermittlungsstellen ist abhängig von der Gesamtzahl der zu übertragenden Pakete. Die Kommunikation zwischen Sender und Empfänger kann entweder verbindungslos (connectionless) oder verbindungsorientiert (connectionoriented) ablaufen. Ein Beispiel für die Paketvermittlung ist die Datenübertragung über TCP/IP im Internet. Dabei ist TCP für das Aufteilen und Zusammensetzen der Daten in Pakete zuständig, während IP für die (ungesicherte) Übertragung verantwortlich ist. IPv4 und IPv6 arbeiteten verbindungslos.

Verbindungsorientierte Kommunikation

Bei einer verbindungsorientierten Kommunikation wird vor der Übertragung einer Nachricht eine logische Verbindung zwischen den Kommunikationspartnern aufgebaut. Diese Art der Nachrichtenübertragung besteht im Prinzip aus drei Phasen: Verbindungsaufbau, Datenübertragung und Verbindungsabbau (vgl.Abbildung 3). Für den Aufbau der Verbindung ist die Angabe der vollständigen Netzadresse des Empfängers notwendig. Es können auch bestimmte Dienstmerkmale abgesprochen werden (z. B. Übertragungsrate eines Modems). Ein Beispiel für eine verbindungsorientierte Kommunikation ist das analoge Fernsprechnetz.

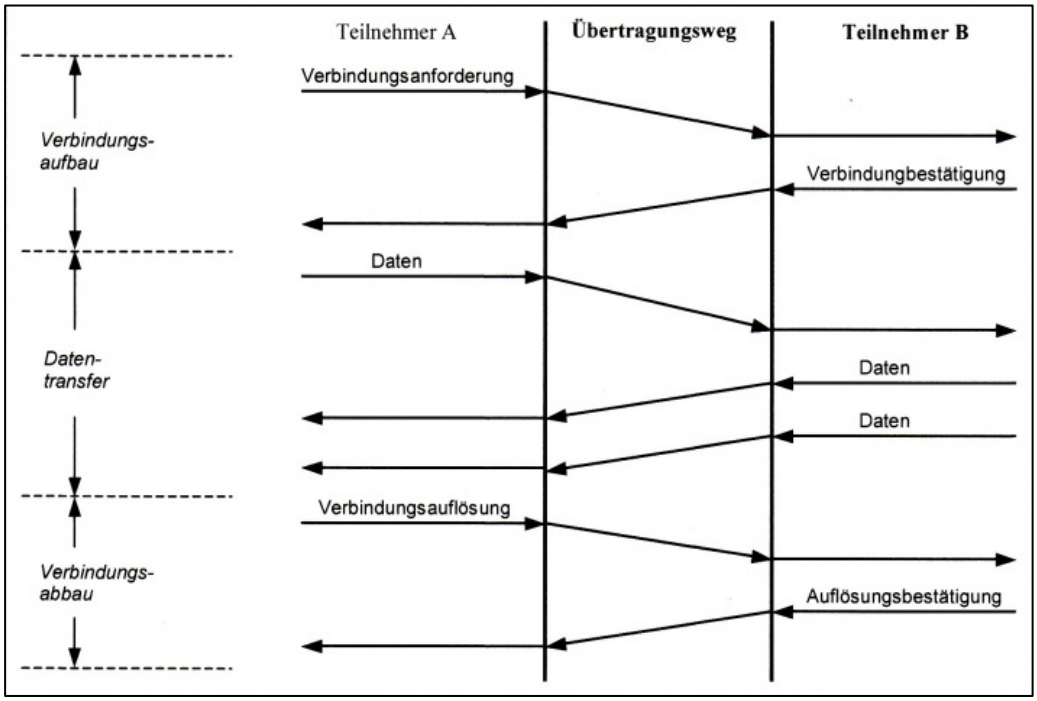


Abbildung 3: Prinzip einer verbindungsorientierten Kommunikation

Verbindungslose Kommunikation

Bei einer verbindungslosen Kommunikation ist jede Nachrichtentransaktion unabhängig von den vorangegangenen oder nachfolgenden. Vor dem Datenaustausch muss keine Verbindung zwischen Sender und Empfänger aufgebaut werden. Aufeinanderfolgende Datenblöcke können über verschiedene Wege zum gleichen Ziel gelangen. Jeder Block benötigt deshalb eine Absender- und Empfängeradresse. Die einzelnen Blöcke können unter Umständen auch in der falschen Reihenfolge ankommen. Ein Beispiel für eine verbindungslose Kommunikation ist die Datenübertragung im Internet.

1.1.2.2 Leitungsvermittlung (Circuit-Switching, Durchschaltevermittlung)

Ein anderes Vermittlungsprinzip wird beim Telefonnetz angewendet. Werden in einem solchen durchschaltevermittelten (circuit-switched) Netzwerk Verbindungen hergestellt, widmet sich dieser Teil des Netzes ausschließlich dieser einen Verbindung. Dadurch sind sehr hohe Datenraten möglich.

1.1.3 Einteilung von Datennetzen

Als Daten- bzw. Rechnernetz wird ein Verbund von Rechnern, Peripheriegeräten und Kommunikationsgeräten bezeichnet, der dem Informationsaustausch zwischen mehreren Kommunikationspartnern dient. Ein solches Netz kann nach Funktionalität, Zugangsart oder verwendeten Techniken klassifiziert werden. Häufig wird eine Einteilung nach der geographischen Ausdehnung vorgenommen:

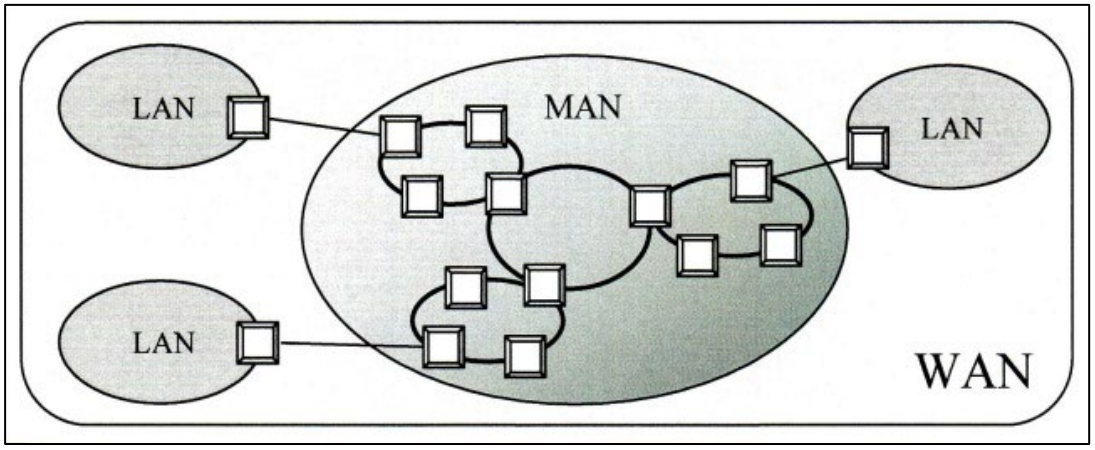


Abbildung 4: LAN, MAN und WAN

Ein Local Area Network (LAN) besteht aus mehreren Computern und Peripheriegeräten, die auf einem begrenzten Raum (maximal einige Kilometer) über Leitungen miteinander verbunden sind. Ein solches Netzwerk kann z. B. innerhalb eines Gebäudes oder in einer kleinen Firma eingerichtet sein. Die Vermittlung erfolgt dezentral, d.h. es gibt keine übergeordnete Instanz, die Vermittlungsaufgaben übernimmt. LANs sind heute in der Regel die Bausteine größerer Systeme. Die Administration und Benutzung obliegt einem definierten Personenkreis.

Metropolitan Area Networks (MANs) sind für die Verbindung von LANs innerhalb eines Großstadtbereiches geschaffen worden. Die mögliche Ausdehnung liegt zwischen der von LANs und WANs. Vorwiegend in diesem Bereich werden Hochgeschwindigkeitstechnologien (Fast Ethernet, FDDI, ATM etc.) entwickelt und getestet. Deshalb werden diese Netze manchmal auch als Hochgeschwindigkeits-LANs bezeichnet. Auch innerhalb eines MAN wird dezentral vermittelt. Die Grenzen zwischen LANs und WANs resultieren aus den technischen Machbarkeiten und verschieben sich mit zunehmendem Fortschritt der Technik.

Wide Area Networks (WANs) sind für die Überbrückung großer Distanzen vorgesehen. Ein WAN besteht in der Regel aus mehreren Teilnetzen, die von verschiedenen Organisationen verwaltet werden und individuelle Dienste innerhalb des Netzwerks anbieten. Adressierung und Streckenführung werden gemeinschaftlich organisiert. Aufgrund der Vielzahl der angeschlossenen Stationen ist eine zentrale Vermittlung erforderlich. Ein LAN kann Teil des WAN sein, wenn die Benutzer über das lokale Netz auf entfernte Rechner im WAN zugreifen können. Das derzeit wohl bekannteste WAN ist das Internet.

1.1.4 Netzwerktopologien

Netze bestehen aus Knoten und Verbindungen zwischen den Knoten. Verbindungen sind Leitungen oder Leitungsbündel und dienen dem Transport von Daten. Knoten können Vermittlungsstellen oder Endgeräte sein. Die drei Grundformen einer Netzwerktopologie sind: Bus-, Stern- und Ringnetz.

Beim Bus-Netz werden alle Computer durch ein einziges Kabel (Bus) verbunden. Um eine Signalreflexion in der Leitung zu verhindern, befindet sich an jedem Kabelende ein Abschlusswiderstand. Es kann immer nur ein Computer Daten auf das gemeinsam genutzte Übertragungsmedium senden („Shared Medium“). Die Signale können von allen Computern im Netz „gehört“ werden. Die Netzwerkleistung hängt sehr stark von der Anzahl der am Bus angeschlossenen Computer ab. Je mehr Computer sich in diesem Netz befinden, desto häufiger kommt es zu Wartezeiten für Rechner, die Daten senden möchten.

Im Ringnetz sind alle Rechner in einem logischen Kreis verbunden. Beim Token-Passing wird ein sogenanntes Token (Sendeberechtigung) von einem Computer zum nächsten geleitet. Wenn ein Computer senden will, modifiziert er das freie Token, versieht die Daten mit einer Empfängeradresse und gibt sie auf den Ring. Der Zielrechner erkennt die Übereinstimmung der Empfängeradresse mit seiner eigenen und liest die Daten. Anschließend schickt er eine Empfangsbestätigung an den Absender. Sobald dieser die Bestätigung erhält, erzeugt er ein neues Token und speist es ins Netz ein.

Ein Stern-Netz besitzt einen zentralen Knoten, mit dem jeder einzelne Computer verbunden ist. Je nach Größe des Netzes können durch Kaskadierung baumartige Strukturen entstehen. Wenn eine Leitung ausfällt, hat diese Unterbrechung nur Einfluss auf das jeweilige Segment. Wenn die zentrale Komponente ausfällt, kommt das gesamte Netzwerk zum Stillstand.

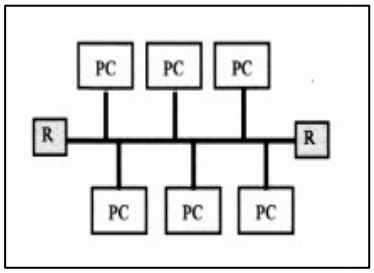


Abbildung 5: Bus-Struktur

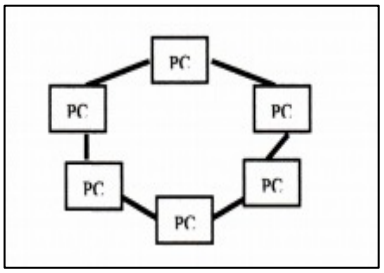


Abbildung 6: Ring-Struktur

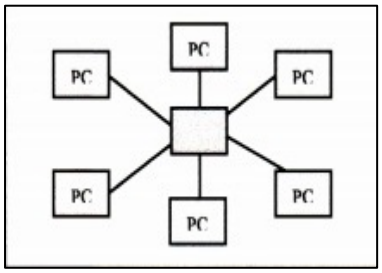


Abbildung 7: Stern-Struktur

In vielen Fällen werden diese Grundformen entsprechend den konkreten Erfordernissen kombiniert. In den letzten Jahren werden einzelne Netze verstärkt über ein sogenanntes Backbone (Rückgrat) zusammengeschlossen. Dieser sehr leistungsfähige Netzstrang verbindet auch Netze unterschiedlicher Topologien. Das heutige Internet ist in erster Linie aus ring- und sternförmigen Teilsystemen verschiedenster Komplexität und Ausdehnung zusammengesetzt.

1.2 Routing

Die Routenwahl ist der Prozess, durch den die optimale Route für eine Kommunikationsverbindung (verbindungsorientiert) bzw. für einen einzelnen Nachrichtenblock (verbindungslos) in einem verteilten Netz beliebiger Komplexität festgelegt wird. Leitweglenkung ist eine Funktion von zentralvermittelten Netzen. Abhängig von der eingesetzten Topologie in einem Netzwerk können Nachrichtenpakete auf verschiedenen Wegen vom Sender zum Empfänger gelangen. Die Hauptaufgabe der Leitweglenkung liegt darin, diejenigen Übertragungsstrecken zu finden, die für die Übermittlung der Daten am besten geeignet sind. Dabei können bestimmte Kriterien vorgegeben werden, wie z. B. Verfügbarkeit, Geschwindigkeit oder Sicherheit.

1.2.1 Repeater, Bridge und Router

Mehrere LAN-Segmente lassen sich durch Repeater, Bridges oder Router zusammenschalten. Diese Komponenten verbinden ein Netzwerk nicht nur physikalisch miteinander, sondern arbeiten auf unterschiedlichen logischen Ebenen. Eine Klassifizierung kann mit Hilfe des OSI-Modells (siehe Kapitel 1.1.1.1) vorgenommen werden.

Repeater

Ein Repeater ist ein Signalregenerator und -verstärker mit Ein- und Ausgangsport (in der Regel für Hin- und Rückweg). Er verbindet zwei physikalische Segmente auf der Bit-Übertragungsschicht (Layer 1). Soll ein Segment erweitert werden, welches das Limit seiner physikalisch erlaubten Ausdehnung (Kabellänge) bereits erreicht hat, wird ein Repeater eingesetzt, um die elektrischen Impulse zu verstärken. Der Repeater ermöglicht keine Trennung der Verkehrslast zwischen den an ihn angeschlossenen Segmenten. Treten auf einem Netzsegment Kollisionen auf, werden diese direkt auf das andere Segment übertragen.

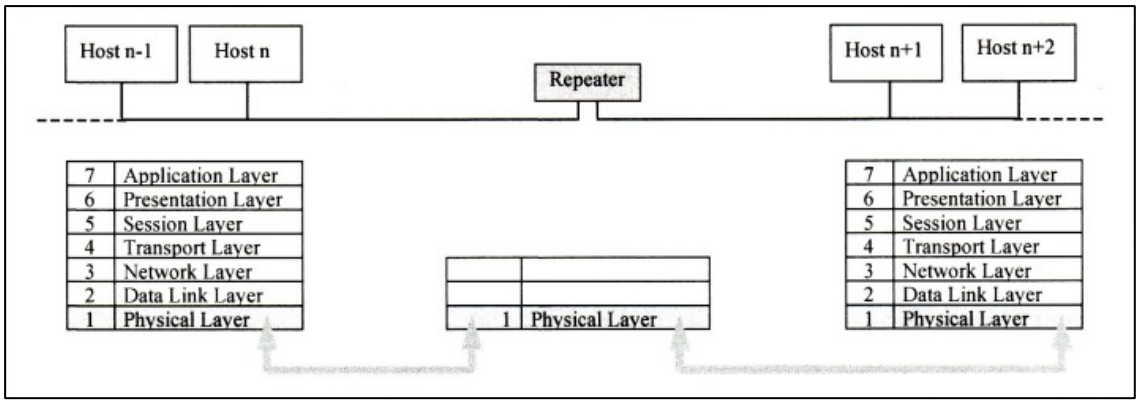


Abbildung 8: Netzwerkverbindung mittels Repeater

Hub

Der Hub ist in seiner Funktionsweise ein Repeater, allerdings hat er im Vergleich zu diesem mehr als zwei Anschlussports. Man kann den Hub daher auch als Multiport-Repeater bezeichnen. Ein eingehendes Netzwerkpaket wird an alle Anschlussports des Hubs weitergeleitet. Somit können alle Geräte, die an dem Hub angeschlossen sind, den Netzwerkverkehr des Hubs „mithören“. Genau wie der Repeater arbeitet der Hub auf Schicht 1 des OSI-Modells. Zieladressen in Form von MAC- oder IP-Adressen sind für ihn irrelevant.

Bridge

Bridges arbeiten auf der Sicherungsschicht (Layer 2) und leiten Pakete entsprechend der im Paketkopf (Header) enthaltenen MAC-Adresse (Hardware-Adresse) weiter. Sie sind unabhängig von höheren Protokollen (z. B. TCP/IP) und können daher Pakete nur anhand von Hardware-Adressen, nicht aber anhand von IP-Adressen zustellen.

Bridges erhöhen die Ausfallsicherheit, da Störungen von der einen Seite der Bridge nicht auf die andere Seite gelangen. Gleichzeitig verbessern sie die Datensicherheit, weil Informationen, die zwischen Adressen auf einer Seite der Bridge ausgetauscht werden, nicht auf der anderen Seite der Bridge abgehört werden können (Passwörter, etc.). Bei durchschnittlich belasteten Netzen kann durch den sinnvollen Einsatz von Bridges die Effizienz des Gesamtnetzwerkes erhöht werden, weil nicht alle Pakete des Gesamtnetzes auf jedem Netzwerkteil übertragen werden müssen. Deshalb sollten Rechnergruppen, die viel miteinander kommunizieren, durch Bridges vom Gesamtnetz getrennt werden.

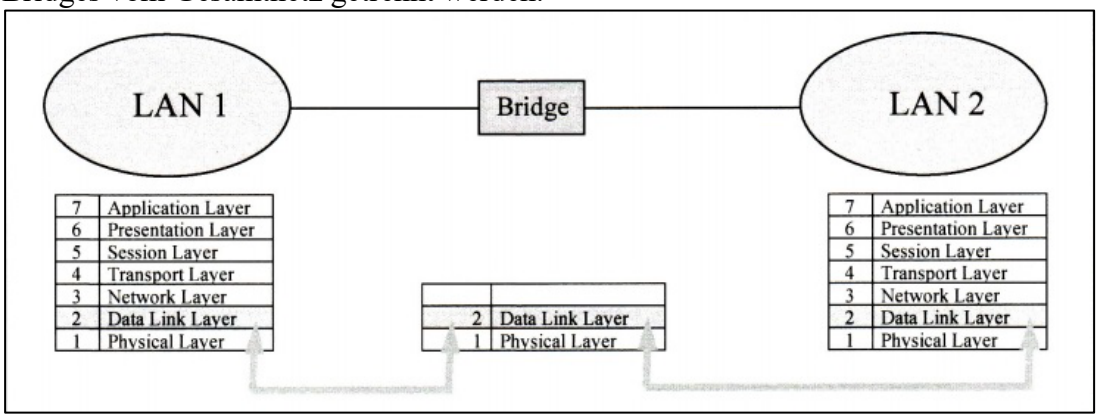


Abbildung 9: Kopplung zweier Netzwerke über eine Bridge

Statische Bridges arbeiten mit einer Tabelle, in der alle in den angeschlossenen Segmenten vorkommenden Adressen enthalten sind. Wenn sich der Standort einzelner Rechner ändert oder neue Rechner ins Segment integriert werden sollen, müssen die Adresstabellen aller beteiligten Bridges geändert werden. Wesentlich flexibler sind intelligente Bridges, die den Verkehr auf den Segmenten analysieren und ihren Adressbestand dynamisch aktualisieren. Eine intelligente Bridge analysiert die Absender- und Empfängeradressen aller ankommenden Datenblöcke. Unbekannte Absenderadressen werden in die eigene Adresstabelle eingetragen. Die Bridge lernt so, an welchem Anschluss jeder bekannte Rechner (MAC-Adresse) zu finden ist. Wenn die Empfängeradresse nicht bekannt oder einem anderen Anschluss der Brücke zugeordnet ist, wird das Paket weitergeschickt. Bei managebaren Bridges kann die Weitergabe von Daten durch zusätzliche Adressfilter eingeschränkt werden.

Switch

Ein Switch ist letztendlich eine Bridge mit mehr als zwei Ports. Eingehende Pakete werden anhand der Zieladresse (MAC-Adresse) nur an den zugehörigen Anschlussport weitergeleitet, an dem sich das Zielgerät befindet. Geräte an anderen Anschlussports bekommen von diesem „fremden“ Datenverkehr nichts mit und ihr Netzsegment bleibt frei (somit ist auch keine Wartezeit oder Kollisionsbehandlung bezüglich des fremden Datenverkehrs notwendig).

Router

Alle Geräte, die die Routenwahl mit Hilfe von IP-Adressen durchführen, werden als (IP)-Router bezeichnet. In der TCP/IP-Welt wurde ursprünglich der Begriff Gateway verwendet. Weil die OSI-Standards mit „Router“ und „Gateway“ zwischen zwei verschiedenen Funktionen unterscheiden, wird in der neueren Literatur in der Regel der Begriff „Router“ benutzt.

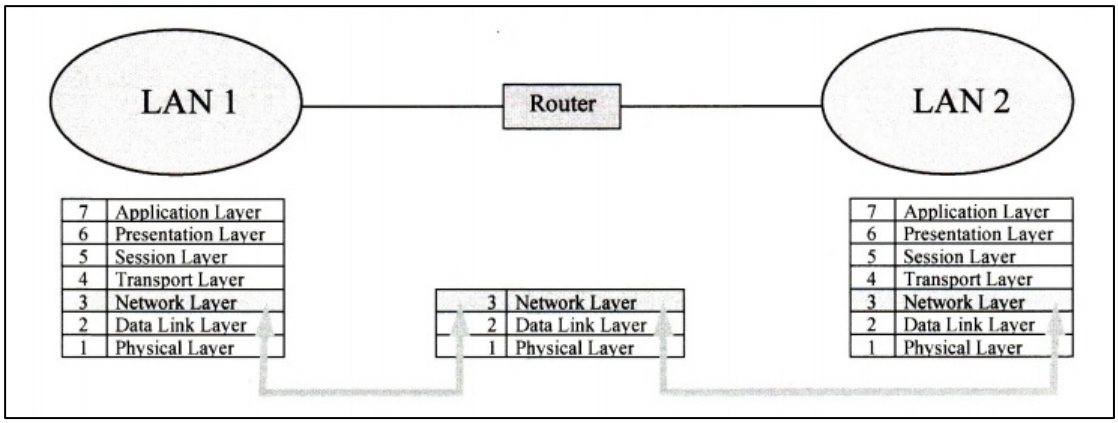


Abbildung 10: Netzwerkverbindung mittels eines Routers

Ein Router arbeitet auf der Vermittlungsschicht (Network Layer) und trennt ein Netz in Bereiche mit verschiedenen IP-Adressbereichen. Die Weiterleitung von Paketen erfolgt anhand dieser IP-Adressen. Router können für ein oder mehrere Netzwerkprotokolle ausgelegt sein und auch Netze unterschiedlicher Topologie verbinden. Da die Verbreitung von heterogenen Netzen stark zunimmt, geht der Trend bei Routern in Richtung Multiprotokoll-Fähigkeit. Weil nicht alle Protokolle geroutet werden können, sind die meisten Router auch in der Lage, Pakete zu „bridgen“. Deshalb ist die Bezeichnung Bridge/Router für solche Geräte präziser. Gegenüber herkömmlichen Bridges gewährleisten Router eine bessere Isolation des Datenverkehrs, da sie z. B. Broadcasts (ein Absender schickt ein Paket an alle möglichen Rechner im gleichen IP-Netz) nicht standardmäßig weiterleiten. Auf der anderen Seite sind die Verzögerungszeiten gegenüber Bridges etwas größer, weil Router jedes IP-Paket vor dem Weiterleiten erst analysieren und anhand ihrer Routing-Tabellen weiterleiten müssen. In extrem weit verzweigten Netzwerken (Internet) werden die Daten schlussendlich jedoch effektiver zum Empfänger geleitet.

1.2.2 Statisches und dynamisches Routing

Die meisten Routing-Algorithmen verwenden Tabellen, die in den einzelnen Hosts (Stationen, Rechnern) geführt und aktualisiert werden. Mit Hilfe dieser Tabellen wird entschieden, auf welchem Weg die Nachricht durch das Netzwerk transportiert werden soll. Die Tabelleneinträge können dabei statisch oder dynamisch sein. Statisches Routing bedeutet, dass alle relevanten Tabelleneinträge durch einen Administrator von Hand vorgenommen werden müssen. Dies ist später auch Ihre Aufgabe in diesem Laborversuch. Diese Methode führt schon bei kleineren Netzen zu einem immensen Arbeitsaufwand, wenn z. B. neue Rechner ins Netz integriert oder einzelne Computer an einem anderen Standort angeschlossen werden sollen. Im Gegensatz dazu werden beim dynamischen Routing die Tabelleneinträge der aktuellen Situation im Netz angepasst (adaptive Routing-Methode). So können z. B. bei hohem Verkehrsaufkommen oder Ausfall eines Netzknotens Überlastsituationen auf einzelnen Leitungsabschnitten vermieden werden. Diese Methode erfordert jedoch einen selbständigen Informationsaustausch aller Stationen.

1.2.3 Ethernet-Adressen

Jedes System besitzt zwei unterschiedliche Adressen: Die Hardware-Adresse (Ethernet- bzw. MAC-Adresse), die auf dem MAC-Layer (Teilschicht der OSI-Schicht 2) jeder Netzkarte fest zugeordnet ist, und die IP-Adresse, die auf der OSI-Schicht 3 frei konfiguriert werden kann.

Ethernet ist ein eingetragenes Warenzeichen der Firma XEROX. Jeder Ethernet-Controller ist durch eine weltweit einmalige Ethernet-Adresse (48-bit Hardware-Adresse; MAC-Adresse) eindeutig identifizierbar. Die Vergabe der Adressbereiche an die Hersteller der Ethernet-Controller erfolgt zentral durch die IEEE. Dafür muss jeder Hersteller von Ethernet-Geräten eine Lizenzgebühr an XEROX zahlen. Ethernet-Adressen werden in der Regel in hexadezimaler Form ausgedrückt. Die Schreibweise ist sechs Blöcke zu je zwei hexadezimalen Ziffern. Die Blöcke werden durch einen Doppelpunkt getrennt.

Beispiel: ab:01:57:c8:23:7f

Im den alten, klassischen Ethernet-System (heutzutage praktisch nicht mehr im Einsatz) kommunizieren die einzelnen Stationen über Koaxialkabel-Bus-Verbindungen miteinander. Es gibt keine zentrale Vermittlungsstelle, sondern eine Aufteilung in einzelne Segmente (Baum ohne Wurzel). Der verwendete Medienzugriffsmechanismus – der von vielen unterschiedlichen Protokollen genutzt werden kann – heißt **CSMA/CD (Carrier Sense Multiple Access / Collision Detection)**. Die einzelnen Stationen prüfen vor einer Übertragung, ob das Übertragungsmedium frei ist. Nur wenn das Ergebnis dieses Tests positiv ausfällt, wird gesendet (CSMA). Wegen der Signallaufzeiten auf dem Medium kann es trotzdem zu Kollisionen kommen. Jede Station beendet ihre Übertragung sofort, wenn sie eine Kollision entdeckt (CD).
CSMA/CD ermöglicht mehreren Rechnern den Zugriff auf ein gemeinsam genutztes Medium (Kabel), allerdings kann dieser Zugriffsmechanismus nur eine beschränkte Anzahl von Knoten und ein Kabelsystem von begrenzter Länge verwalten. Diese Beschränkungen sind in den physikalischen Eigenschaften des Netzes begründet und von den Protokollen unabhängig, daher nicht TCP/IP-spezifisch. Bei der Planung eines Netzes müssen diese Restriktionen berücksichtigt werden.

Aufteilung der zu sendenden Daten
Die zu übertragenden Daten werden in Blöcke (Pakete) aufgeteilt. Jedem Block wird ein sogenannter Header vorangestellt, der nähere Informationen zu den Daten enthält (vgl. OSI-Modell). Jede Kommunikationsschicht auf der Absenderseite fügt einen eigenen Header hinzu. Am Zielort werden diese Header von den jeweiligen Kommunikationsschichten gelesen. Die im Paket enthaltenen Daten (inkl. der Header der höheren Schichten) werden der nächsthöheren Schicht übergeben.

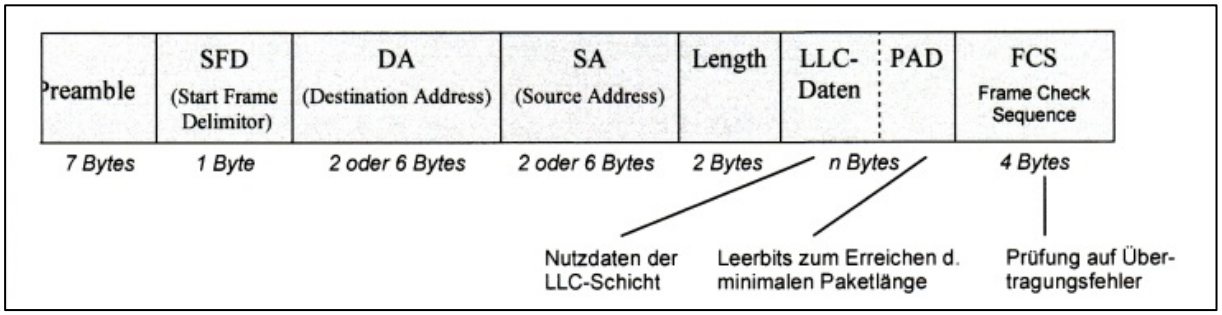


Abbildung 11: Aufbau von Paketen auf der MAC-Schicht bei IEEE 802.3

Die Größe von Ethernet-Rahmen liegt im Bereich von minimal 64 Bytes bis zu 1518 Bytes (Rahmeninhalt ohne Präambel). Die maximale Größe stellt zugleich auch die maximale Datenmenge dar, die in einem Rahmen übertragen werden kann. Aufgrund der Felder des Rahmenkopfes, der CRC-Summe und der Verwaltung durch die in höheren Schichten angesiedelten Protokolle IP und TCP bzw. UDP stehen für die Anwendungsdaten weniger als 1518 Bytes zur Verfügung.

1.2.4 IPv4

Ein Host kann durch seinen Namen oder seine IP-Adresse identifiziert werden. Eine IPv4-Adresse muss innerhalb eines IPv4-Netzes eindeutig sein. Ist der Host lediglich in ein separates Netzwerk eingebunden (z.B. in einer Abteilung eines Unternehmens oder in einem Heimnetz), darf die Adresse zumindest innerhalb dieses IP-Netzes nur einmal vergeben werden. Ist der Host hingegen direkt (ohne Router) mit dem Internet verbunden, muss diese Adresse im gesamten Internet eindeutig sein. Eine IPv4-Adresse ist eine logische 32-Bit-Adresse, die einem IP-Host zugewiesen wird. Jede IPv4-Adresse besteht aus der Netzwerk-ID (Net-ID) und der Host-ID. Net-ID und Host-ID sind in ihrer Länge (Anzahl der Bits) variabel, in Summe ergeben sie gemeinsam aber immer 32 Bit. Die Net-ID identifiziert die Menge aller möglichen Hosts, die sich im gleichen IP-Subnetz befinden. Alle Hosts im gleichen IP-Subnetz haben also eine identische Net-ID. Die Host-ID wiederum kennzeichnet einen bestimmten Host in einem IPv4-Subnetz.

Beispiel mit Net-ID Länge **24** :

	24 Bits der Net-ID	8 Bits der Host-ID
a)	0101 0011 0110 1011 0000 0101	1111 0110
b)	0101 0011 0110 1011 0000 0101	0101 1011
c)	0101 0011 0110 1011 0000 1010	0101 1100

Bei einer Net-ID Länge von 24 gehören die Adressen a) und b) zum gleichen IP-Subnetz, weil ihre Net-IDs eine identische Bitfolge haben. Die Adresse c) gehört zu einem anderen IPv4-Subnetz. Wäre die Net-ID nur 20 Bits lang, würden alle drei Adressen zum gleichen IPv4-Subnetz gehören.
Die Bit-Schreibweise ist bei IPv4-Adressen zeitintensiv und unüblich. Durchgesetzt hat sich die Dotted Decimal Schreibweise (vier dezimal geschrieben Oktetts von 0 bis 255, jeweils durch Punkt getrennt). Die Beispiele von oben in Dotted Decimal Schreibweise:

a) 83.107.5.246 b) 83.107.5.91 c) 83.107.10.92

Bei der Vergabe von IPv4-Adressen sind einige Besonderheiten zu beachten. So haben viele Adressen oder Adressbereiche besondere Bedeutungen (unvollständige Auflistung):

Tabelle 2: Besondere Adressen bei der IP-Adressvergabe

Adressen	Besonderheiten
10.0.0.0 bis 10.255.255.255 172.16.0.0 bis 172.31.255.255 192.168.0.0 bis 192.168.255.255	Reserviert für privaten/internen Gebrauch. Diese Bereiche werden im öffentlichen Internet nicht weitergeleitet und dürfen dort auch nicht auftauchen. Jeder darf diese Adressen in seinem privaten LAN frei vergeben. Hosts mit diesen IP-Adressen dürfen nur durch einen IP-Router mit dem öffentlichen Internet verbunden werden.
127.0.0.0 bis 127.255.255.255	Reserviert für „loopback“: Pakete verlassen den Host nicht, sondern werden an ihn selbst zurück geschickt.
224.0.0.0 bis 239.255.255.255	Reserviert für „Multicast“: Verteilen von Daten von einem Sender an viele Empfänger gleichzeitig (Beispiel: IP-TV).
255.255.255.255	„Limited Broadcast“: Rundruf an alle möglichen Hosts im gleichen IP-Subnetz. Verbleibt im Adressbereich des Absenders und wird standardmäßig nicht in andere IP-Subnetze weitergeleitet.

1.2.4.1 IPv4-Subnetze und Netzmasken

Das Internet besteht aus vielen einzelnen IP-Netzwerken, die wiederum in Subnetz-Strukturen untergliedert sein können. Für die Einrichtung von Subnetzen gibt es verschiedene Gründe:

- Verbindung von Netzbereichen mit unterschiedlichen Technologien (Ethernet, Token-Ring etc.).
- Sicherheitsrelevante Auftrennung in mehrere, kleinere Subnetze, die nicht oder nur begrenzt aufeinander zugreifen sollen/dürfen (Trennung durch Router mit Firewalls).
- Technologisch bedingte Ausdehnungsgrenzen wie maximale Knotenzahl (Token-Ring) oder maximale Segmentlänge (Ethernet) sind erreicht.
- „Bandbreitenreservierung“ durch Einrichtung kleinerer Teilnetze mit geringer Teilnehmerzahl.

IP-Subnetze werden mit Hilfe von Subnetz-Masken eingerichtet. Die Subnetz-Maske ist wie die IP-Adresse ebenfalls eine 32-Bit-Ziffer, die einem Netzwerkgerät stets zusammen mit der IP-Adresse zugewiesen wird. Sie gibt an, wie lang die Net-ID ist, also wie viele mögliche Hosts sich innerhalb dieses Subnetzes maximal befinden können. Die Net-ID wird sozusagen „maskiert“. Die Größe des IP-Subnetzes wird durch die Anzahl der Bits mit dem Wert Null in der Subnetz-Maske bestimmt. Man beginnt von rechts ausgehend (LSB) mit Nullen, bis die benötigte Menge an Hosts für Anzahl der Nullen das Subnetz erreicht ist, also 2 mögliche IP-Adressen. Der Rest der Subnetz-Maske wird mit Einsen aufgefüllt (nur ein einziger Wechsel von 0 auf 1 findet statt). Die Subnetz-Maske 0.0.0.0 ist ein Sonderfall und bedeutet sinngemäß „alle anderen“, da sie den gesamten möglichen Adressbereich umfasst (siehe Tabelle 3). Sie wird häufig als sogenannter Default-Gateway für lokale Netze ins Internet verwendet („ leite alles ins Internet, was nicht zum eigenen lokalen Subnetz gehört “).

Tabelle 3: Übersicht aller möglichen Subnetz-Masken und zugehöriger Subnetz-Größen

Dotted Decimal	Net -ID Bits	Host -ID Bits	Anzahl IP-Adressen	Dotted Decimal	Net -ID Bits	Host -ID Bits	Anzahl IP-Adressen
0.0.0.0	0	32	2^{32}=4,3 Mrd.				
128.0.0.0	1	31	2^{31}=2.2 Mrd.	255.255.128.0	17	15	2^{15}=32768
192.0.0.0	2	30	2^{30}=1,1 Mrd.	255.255.192.0	18	14	2^{14}=16384
224.0.0.0	3	29	2^{29}=537 Mill.	255.255.224.0	19	13	2^{13}=8192
240.0.0.0	4	28	2^{28}=268 Mill.	255.255.240.0	20	12	2^{12}=4096
248.0.0.0	5	27	2^{27}=134 Mill.	255.255.248.0	21	11	2^{11}=2048
252.0.0.0	6	26	2^{26}=67 Mill.	255.255.252.0	22	10	2^{10}=1024
254.0.0.0	7	25	2^{25}=34 Mill.	255.255.254.0	23	9	2^{9}=512
255.0.0.0	8	24	2^{24}=16,7 Mill.	255.255.255.0	24	8	2^{8}=256
255.128.0.0	9	23	2^{23}=8.4 Mill.	255.255.255.128	25	7	2^{7}=128
255.192.0.0	10	22	2^{22}=4.2 Mill.	255.255.255.192	26	6	2^{6}=64
255.224.0.0	11	21	2^{21}=2.1 Mill.	255.255.255.224	27	5	2^{5}=32
255.240.0.0	12	20	2^{20}=1048576	255.255.255.240	28	4	2^{4}=16
255.248.0.0	13	19	2^{19}=524288	255.255.255.248	29	3	2^{3}=8
255.252.0.0	14	18	2^{18}=262144	255.255.255.252	30	2	2^{2}=4
255.254.0.0	15	17	2^{17}=131072	255.255.255.254	31	1	2^{1}=2
255.255.0.0	16	16	2^{16}=65536	255.255.255.255	32	0	2^{0}=1

Tabelle 4: Alle möglichen Bit- und Dezimal-Schreibweisen eines Oktetts in einer Subnetz-Maske

Bit-Schreibweise	Dezimal-Schreibweise
0000 0000	0
1000 0000	128
1100 0000	192
1110 0000	224
1111 0000	240
1111 1000	248
1111 1100	252
1111 1110	254
1111 1111	255

Typische und häufig genutzte Vertreter von Subnetzen sind die Class A-, B- und C-Netze, da mit ihnen besonders einfach gerechnet werden kann (die Bits eines Oktetts sind entweder komplett 0 oder 1).

- 255. 0. 0. 0: 2^{24} = 16,7 Millionen mögliche IP-Adressen Class A Netz
- 255. 255. 0. 0: 2^{16} = 65536 mögliche IP-Adressen Class B Netz
- 255. 255. 255. 0: 2^{8} = 256 mögliche IP-Adressen Class C Netz

1.2.4.2 Broadcast und Net-ID Adresse bei IPv4

Zwei Adressen innerhalb eines IP-Subnetzes sind stets reserviert und dürfen nicht belegt werden:

- Netzwerk-ID / Net-ID: Kleinste mögliche Adresse; alle Host-Bits sind 0.
- Broadcast Adresse: Größte mögliche Adresse; alle Host-Bits sind 1.

Somit stehen beispielsweise im Class C Netz nur 254 belegbare Adressen zur Verfügung.
Achtung: Broadcast bzw. Net-ID haben nicht zwingend immer die Werte 255 bzw. 0.

1.2.4.3 Routing: Gleiches oder fremdes IPv4-Subnetz?

Ein IP-Host will einen anderen IP-Host erreichen. Durch bitweisen Vergleich der IP-Adressen mit der Subnetz-Maske des Absenders wird zunächst überprüft, ob der Empfänger im gleichen Subnetz ist.

		Bits der Net-ID	Bits der Host-ID
Absender IP:	10.0.3.16	0000 1010	0000 0000 0000 00 11 0001 0000
Absender Subnetz-Maske:	255.255.252.0	1111 1111	1111 1111 1111 11 00 0000 0000
Empfänger IP:	10.0.4.156	0000 1010	0000 0000 0000 01 00 1001 1100

Die Empfänger-IP unterscheidet sich in diesem Beispiel an einer Bit-Position, die durch die Subnetz-Maske mit einer 1 maskiert ist. Der Ziel-Host ist daher nicht im gleichen IP-Netz und kann somit nur über einen IP-Router (bzw. Gateway) erreicht werden. IP-Hosts im gleichen IP-Subnetz dürfen sich nur an Bit-Positionen unterscheiden, an denen die Subnetz-Maske eine 0 hat (Host-Bits).

Die anschließende Zustellung eines IP-Pakets in einem IP-basierten Netz zeigt Abbildung 13 in vereinfachter Form. Damit ein IP-Paket nicht in eine unendliche Schleife läuft, enthält es im Header einen Zähler, der beim Durchlauf durch einen IP-Router um 1 verringert wird (TTL: Time To Live). Erreicht der Zähler schließlich 0, so wird das Paket verworfen.

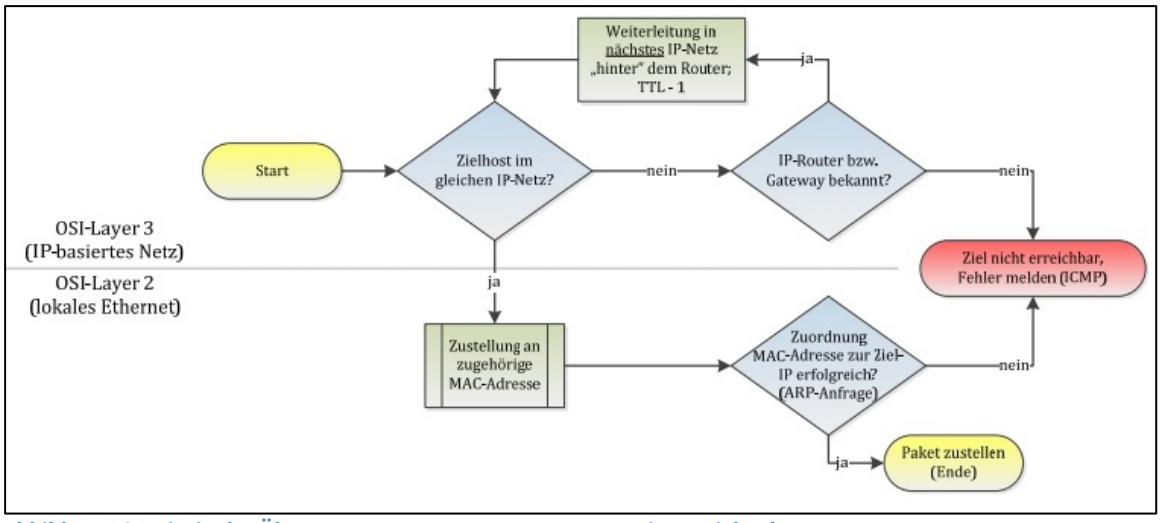


Abbildung 12: Prinzip der Übertragung von IP-Datagrammen zu einem Zielrechner

1.2.4.4 Beispiel für die Einrichtung eines Subnetzes bei IPv4

Einer Firma wurde der Bereich 195.80.16.0 und 195.80.17.0 zugeteilt (512 Adressen). Auf IP-Schicht arbeitende Geräte behandeln normaler Weise die vollständigen IP-Adressen. Da sich hier die beiden Adressen aber erst ab dem 3. Oktett unterscheiden, beschränken wir unsere Betrachtungen auf diesen Bereich (siehe Abbildung 13).

	dezimal	binär
3. Oktett der IP-Adresse 195.80.16.0:	16	0001 0000
3. Oktett der IP-Adresse 195.80.17.0:	17	0001 0001
		---
3. Oktett der Netzmaske:	254	1111 1110

Abbildung 13: Ermittlung der zugehörigen Netzmaske für die vorgegebenen Netzwerkadressen

Ermittlung der Subnetz-Maske für das gesamte Firmennetz
Die beiden Oktette der zwei IP-Adressen werden von links ausgehend bitweise verglichen. Für jede übereinstimmende Bitposition wird in der Subnetz-Maske eine 1 notiert. Trifft man schließlich auf die erste nicht übereinstimmende Bitposition (hier Position 8), wird der Rest der Subnetz-Maske mit 0 aufgefüllt. Es ergibt sich also der Wert 11111110 bzw. 254 für dieses Oktett. Die gesamte Subnetz-Maske ergibt sich für das Firmennetz somit zu 255.255.254.0, oder in Bit-Schreibweise 11111111.11111111.11111110.00000000. Es stehen 9 Bits für die Vergabe von IP-Adressen der Hosts in diesem Subnetz zur Verfügung (abzüglich Broadcast und Net-ID).

CIDR-Notation (Classless Inter-Domain Routing)
Eine schnellere und kompaktere Schreibweise eines IP-Subnetzes ist die CIDR-Schreibweise. Dies ist die Net-ID (alle Host-Bits sind Null, also in diesem Beispiel die letzten 9 in der IP-Adresse) gefolgt von der Anzahl der maskierten Bits der Net-ID. Das hier vorliegende Firmennetz aus dem Beispiel wäre in dieser Notation:

195.80.16.0/23 → Die Net-ID umfasst 8 + 8 + 7 + 0 = 23 Einsen.

Die CIDR-Notation verrät uns stets folgende Informationen über ein IP-Subnetz:

- A.B.C.D/X (32-X)
    - Größe des IP-Subnetzes: $2^{(32-X)}$ mögliche IP-Adressen, abzgl. Broadcast und Net-ID
    - Subnetz-Maske: X Einsen gefolgt von (32-X) Nullen
    - Net-ID: Niedrigste mögliche IP-Adresse A.B.C.D, alle Host-Bits sind Null

Das Netz soll nun intern entsprechend Abbildung 14 in drei Bereiche aufgeteilt werden.

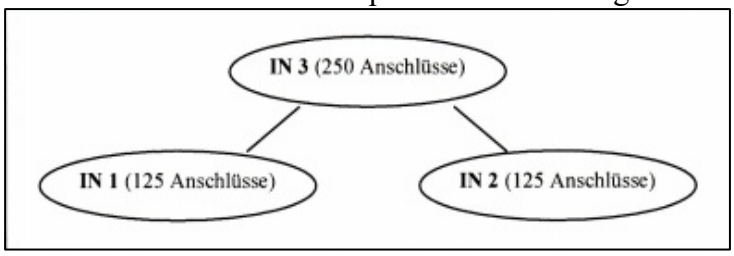


Abbildung 14: Aufteilung des Firmennetzes in drei Subnetze

Bei einem Class C Netzwerk (CIDR-Notation /24) steht das 4. Oktett (8 Bit) für Hostadressen zur Verfügung (256 – 2 = 254 Adressen). Dies ist ausreichend für Subnetz IN 3. Zugeordnet wird in diesem Fall der Bereich 195.80.16.0 bis 195.80.16.255.

Für IN 1 und IN 2 reichen bereits 7 Bit für Hostadressen (2^7 – 2 = 126). Die notwendige Subnetz-Maske lautet 255.255.255.128 (CIDR-Notation /25). Für IN 1 und IN 2 werden folgende Bereiche zugeordnet:

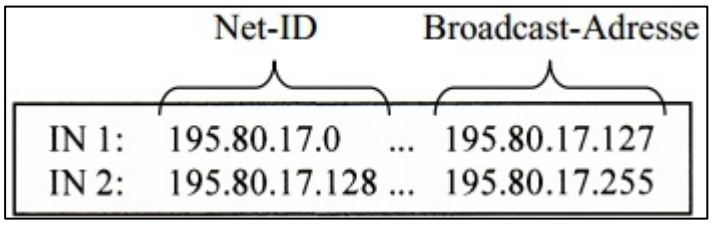


Abbildung 15: IP-Adressbereiche für IN 1 und IN 2

Die Net-ID eines IP-Netzes lässt sich mit Hilfe der „bitweise-AND“-Operation zwischen IP-Adresse und Subnetz-Maske sehr einfach ermitteln. Die folgenden Beispiele sollen dies verdeutlichen.

Beispiel 1:
Ein Rechner aus Subnetz IN 1 mit der IP-Adresse 195.80.17.2 und Subnetzmaske 255.255.255.128:
Die „bitweise-AND“-Operation liefert 0, also ist die Net-ID 195.80.17.0.

	dezimal	binär
4. Oktett der IP-Adresse:	2	0000 0010
4.Oktett der Subnetzmaske:	128	1000 0000
		---------- (bitwise AND)
4. Oktett der Netzwerk-Adresse:	0	0000 0000

Abbildung 16: Ermittlung der Netzwerkadresse für Subnetz IN 1

Beispiel 2:

Ein Rechner aus Subnetz IN 2 mit der IP-Adresse 195.80.17.130 und Subnetzmaske 255.255.255.128:
Die „bitweise-AND"-Operation liefert 128, also ist die Net-ID 195.80.17.128.

	dezimal	binär
4. Oktett der IP-Adresse:	130	1000 0010
4.Oktett der Subnetzmaske:	128	1000 0000
		---------- (bitwise AND)
4. Oktett der Netzwerk-Adresse:	128	1000 0000

Abbildung 17: Ermittlung der Netzwerkadresse für Subnetz IN 2

Schlussendlich erhalten wir gemäß obiger Forderung für das interne Firmennetz in diesem Beispiel folgende IP-Bereiche für die Subnetze IN 1 bis IN 3:

- 195.80.16.0/24: 254 mögliche Hosts im Netz IN 3
- 195.80.17.0/25: 126 mögliche Hosts im Netz IN 1
- 195.80.17.128/25: 126 mögliche Hosts im Netz IN 2

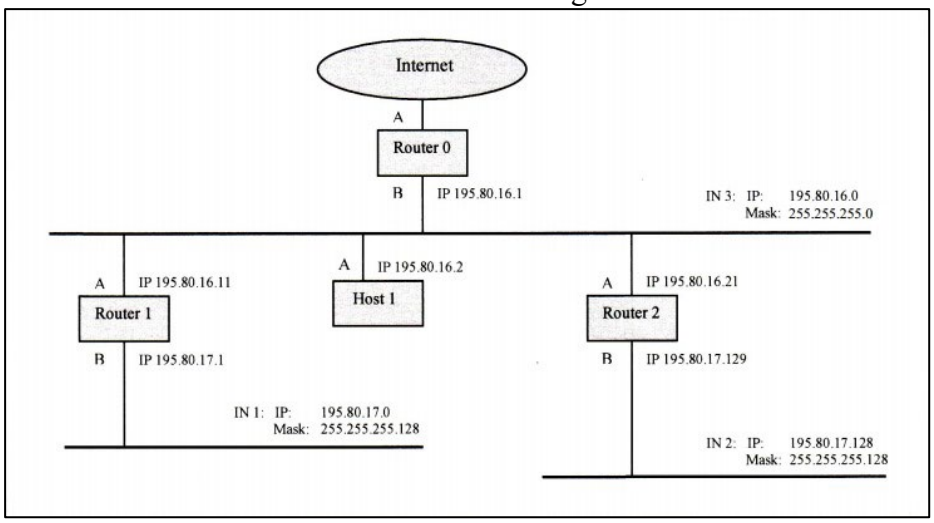


Abbildung 18: Adressen und Masken für ein Netzwerk mit Subnetzen

Für die korrekte Leitweglenkung sind entsprechende Einträge in den Routing-Tabellen notwendig. Einige Routingtabellen für dieses Netzwerk sind im Anhang abgebildet.
Hinweise zu den Routingtabellen:
Ein IP-Host in einem IP-Netz „sieht" nur die Adressen derjenigen Netzwerkgeräte, mit denen er direkt verbunden ist. Will der Rechner ein Netzwerkgerät außerhalb seines eigenen IP-Subnetzes erreichen, muss dem Rechner mitgeteilt werden, welche IP-Adresse für die Weiterleitung in das andere IP-Subnetz benutzt werden soll (also die IP-Adresse des Routers/Gateways). Für eine Weiterleitung muss nur der nächstgelegene Router als Gateway in das gewünschte Zielnetz angegeben werden (also eine Adresse aus dem eigenen Subnetz).

Beispiel:

Ein beliebiger Host (z.B. 195.80.17.25) in Netzwerk IN 1 benötigt nur Kenntnis über seinen nächstgelegenen Router mit der Adresse 195.80.17.1, um die Netzwerke IN 2 oder IN 3 erreichen zu können. Nur diesen Router kann er direkt erreichen. Der Host braucht keine Kenntnis über die anderen Router in den anderen IP-Subnetzen. Die Weiterleitung geschieht „hop by hop“ (vergleiche Abbildung 12). Das heißt, jeder der Router muss auch immer nur seine nächste Station zur Weiterleitung kennen.

1.2.5 IPv6

IPv4 wurde im Jahr 1981 im „Request for Comments“ (RFC) 791 definiert und seitdem nicht mehr grundlegend geändert. Es wurde nach den damaligen Kenntnissen entwickelt und hält heutigen Anforderungen nur noch teilweise stand. Eines der Hauptprobleme ist das rasante Wachstum des Internet und der damit verbundene Adressbedarf. IPv4 ermöglicht theoretisch 4.294.967.296 Adressen von denen durch einige Restriktionen nur etwa einige hundert Millionen Adressen genutzt werden können. Dieser Adresspool reicht schon für das Internet von heute nicht mehr aus. Ein weiterer Nachteil von IPv4 ist die mangelnde Informationssicherheit, die nur durch optionale Nutzung von Sicherheitsprotokollen wie IPsec gewährleistet werden kann. Zudem sind seit dem Beginn des Internets Echtzeitübertragungen wie Voice over IP oder Videotelefonie immer wichtiger geworden. Hierfür wird eine hochwertige Priorisierung von Echtzeitdatenströmen benötigt, die IPv4 nur teilweise erlaubt.

Alle diese Nachteile haben zu Entwicklung von IPv6, der neusten Version des Internet-Protokolls geführt. IPv6-Adressen bestehen aus 128 Bits (16Bytes) und erlaubt damit theoretisch $3{,}4*10^{38}$ oder auch 340.282.366.920.938.463.463.374.607.431.768.211.456 Adressen. Anders ausgedrückt würden für jeden Quadratmillimeter der Erde 667 Billiarden Adressen zur Verfügung stehen. IPv6 bietet zudem einen IPsec-Header zum Schutz der Nutzlast und eine deutlich bessere Unterstützung von Datenpriorisierung für Echtzeitanwendungen. Ebenfalls wird eine Autokonfiguration in lokalen Netzwerken unterstützt. Somit lassen sich nach Aktivierung von IPv6 alle Hosts im lokalen Netz sofort erreichen.

Im Gegensatz zu IPv4 wurde der Header bei IPv6 auf das absolut notwendige gekürzt und hat eine feste Länge. Dies erlaubt eine schnellere Bearbeitung der Pakete. Sollen weitere Funktionen, wie zum Beispiel eine Verschlüsselung der Nutzlast stattfinden, wird dies über weitere optionale Header erreicht. Diese Header haben jeweils nur eine Funktion und werden nur dann angehängt, wenn sie tatsächlich gebraucht werden. Abbildung 19 stellt den IPv6 Basis-Header dar.

<table>
<tr><td>Version (4Bit)</td><td>Class (8Bit)</td><td colspan="3">Flow-Label (20Bit)</td></tr>
<tr><td colspan="3">Payload Legth (16Bit)</td><td>Next (8Bit)</td><td>Hop-Limit (8Bit)</td></tr>
<tr><td colspan="5">Source Address (128Bit)</td></tr>
<tr><td colspan="5">Destination Address (128Bit)</td></tr>
</table>

Abbildung 19: IPv6-Header

Version kennzeichnet unterschiedliche Versionen von IP. So enthält das Versionsfeld bei IPv6 immer 6.

Durch Class wird die Verkehrsklasse festgelegt. Die Verkehrsklasse macht eine Aussage über die Art der Nachricht und insbesondere die Priorität. Pakete mit hoher Priorität werden bevorzugt behandelt.

Mit Flow-Labels können bei IPv6 Datenströme gekennzeichnet werden. Wird ein Paket mit einem Flow-Label ungleich 0 (0 entspricht kein Flow-Label) bearbeitet, kann der IPv6-Router dies in einer gesonderten Tabelle speichern, in der die Absender-, Zieladresse und das Flow-Label abgelegt werden. Kommt nun ein weiteres Paket desselben Typs an, kann es direkt, ohne jegliches Software-Routing, weitergeleitet werden. So können Echtzeitsitzungsströme wie bei Voice over IP effizienter vermittelt werden.

Die Payload-Length gibt die Länge des restlichen Paketes (ohne den Basis-Header) an. Durch die 16 Bit entsteht eine maximallänge von 65.536 Bytes. Soll ein Paket länger sein, muss ein Optionsheader verwendet werden.

Im Feld Next wird, sofern vorhanden, der Typ des nächsten Optionsheaders angegeben. Falls es keinen weiteren Optionsheader gibt, wird der Typ der Nutzlast angegeben (z.B.: UDP oder TCP).

Das Hop-Limit gibt an, nach wie vielen Durchgängen durch einen Router das Paket vernichtet werden soll.

1.2.5.1 Adressdarstellung bei IPv6

Bei IPv6 wurde eine hexadezimale Schreibweise gewählt um die 128-Bit langen Adressen möglichst kurz darzustellen. Für die Lesbarkeit werden immer Gruppen aus 2 Bytes gebildet und durch einen Doppelpunkt getrennt.

Beispiel: 4264:0000:0000:0000:0000:0001:F37E:0001

Führende Nullen können ausgelassen werden.

Beispiel: 4264:0:0:0:0:1:F37E:1

Einmal innerhalb einer IPv6-Adresse können aufeinanderfolgende Nullen durch zwei Doppelpunkte ersetzt werden. Dies gilt ebenfalls für den Beginn der IPv6-Adresse.

Beispiel: 4264::1:F37E:1

Auch bei IPv6 wird die Adresse grundsätzlich in zwei Teile aufgeteilt, von denen der erste das sogenannte Präfix ist. Er wird wie bei IPv4 mit der mit CIDR eingeführten Notation dargestellt.

Beispiel: 4264::1:F37E:1/40

Der so definierte Netzteil der IPv6-Adresse wird Präfix genannt.

1.2.5.1 Adresstypen bei IPv6

IPv6 enthält drei unterschiedliche Adresstypen:

Unicast-Adressen:
Diese Adressen können genau einem Netzwerkcontroller zugeordnet werden. Pakete, die an Unicast-Adressen geschickt werden, haben demnach nur genau ein Ziel.

Multicast-Adressen:
Adressen dieses Typs beziehen sich auf eine Gruppe von Netzwerkcontrollern. Pakete mit derartigen Adressen werden an alle Ziele zugestellt, die dieser Gruppe angehören. Multicast-Adressen werden hauptsächlich von bestimmten Netzwerkdiensten für die Kommunikation mit bestimmten Hostgruppen verwendet, wobei diese gezielt adressiert werden.

Anycast-Adressen:
Adressen dieses Typs beziehen sich ebenfalls auf eine Gruppe von Controllern. Pakete mit einer derartigen Adresse werden zu dem Mitglied der Gruppe gesendet, welches dem Absender am nächsten ist. Anycast-Adressen werden verwendet, damit Hosts Informationen zu Servern schneller abrufen können, die im angegebenen Netzwerkbereich bestimmte Dienste anbieten. Sämtliche Server desselben Typs verfügen über dieselbe Anycast-Adresse. Wann immer ein Host einen Dienst anfordert, erhält er eine Antwort von dem vom Routing-Protokoll ermittelten nächstgelegenen Server. Wenn dieser Server aus irgendeinem Grund nicht erreichbar ist, wählt das Protokoll automatisch den zweitnächsten Server, dann den dritten usw. aus.
Anhand der ersten Bits einer IPv6-Adresse kann man bestimmen um welche Adressart es sich handelt. Tabelle 5 stellt einige IPv6-Präfixe dar.

Tabelle 5: IPv6-Präfixe

Präfix	Definition
0:0:0:0:0:ffff::/96	IPv4 mapped (abgebildete) IPv6-Adressen. Die letzten 32 Bits enthalten die IPv4-Adresse. Ein geeigneter Router kann diese Pakete zwischen IPv4 und IPv6 konvertieren und so die neue mit der alten Welt verbinden.
2000::/3	globale Unicast-Adressen
FE80::/10	Link-local-Adressen. Dieser Adresstyp ist ausschließlich für den Gebrauch im lokalen Subnetz bestimmt. Router dürfen Pakete mit solcher Ziel- oder Quelladresse nicht an das Internet oder andere Subnetze weiterreichen. Diese Art Adresse wird von den Autokonfigurationsmethoden verwendet, um Hosts im selben Subnetz anzusprechen. Nach dem Präfix folgt nur noch die Schnittstellen-ID der Netzwerkkarte (siehe Abbildung 20).
FC00::/8	Unique-local-Adressen. Pakete mit diesem Adresstyp dürfen zwischen einzelnen Subnetzen geroutet werden, aber nicht außerhalb einer Organisation ins Internet gelangen. Solche Adressen werden für Intranets eingesetzt und sind ein Äquivalent zu den privaten IPv4-Adressen (beispielsweise 192.168.x.x). Neben einem definierten Präfix (FC00::/8) und der Interface-ID enthalten diese Adressen ein 16-Bit-Feld, in dem die Subnetz-ID kodiert ist. Der Rest wird mit Nullen aufgefüllt (siehe Abbildung 22).
FF00::/8	Multicast-Adressen

Mit IPv6 kann ein Netzwerkcontroller mehrere IP-Adressen erhalten. Somit kann er mehreren Netzen angehören. Ein besonderer Vorteil ist, dass mithilfe der MAC-Adresse und einem bekannten Präfix die erste Adresse vollautomatisch konfiguriert wird, sodass gleich nach Aktivierung von IPv6 alle Hosts im lokalen Netz über verbindungslokale (link-local) Adressen erreichbar sind. Durch die MAC-Adresse als Bestandteil der IP-Adresse ist jede dieser Adressen global eindeutig.
Unique-local-Adressen hingegen müssen manuell konfiguriert werden und können mit den privaten Adressen von IPv4 verglichen werden.

1.2.5.2 Routing in IPv6

Wie bei IPv4 wird auch bei IPv6 eine lokale Routing-Tabelle genutzt um IP-Pakete korrekt weiterzuleiten. Diese Tabelle wird automatisch bei Initialisierung der ersten IPv6-Schnittstelle angelegt und dann je nach Konfiguration und Netzwerkverkehr erweitert. Die Routing-Tabellen enthalten Informationen über die IPv6-Präfixe, und wie diese erreicht werden können. Sobald also ein IPv6-Paket an ein unbekanntes Ziel weitergeroutet werden soll, werden anhand der Routing-Tabelle die „Next-Hop-Adress“ und das zu nutzende Interface herausgesucht. Liegt die Zieladresse im selben Subnetz, ist die „Next-Hop-Adress“ gleich der Zieladresse. Liegt die Zieladresse in einem anderen Subnetz, ist die „Next-Hop-Adress“ in der Regel die lokale Adresse des nächsten Routers.
Nach der Weiterleitung des Paketes wird bei IPv6 ein Eintrag im „Destination Cache“ angelegt. Folgen nun weitere Pakete mit derselben Ziel Adresse, wird das Paket direkt nach Informationen des „Destination Cache“ weitergeleitet ohne erneut die Routing-Tabelle auszuwerten.

1.2.5.2 Beispieleinrichtung einer IPv6-Umgebung

IPv6 konfiguriert automatisch für jede Netzwerkschnittstelle eine verbindungslokale IPv6-Adresse (link-local-address). Diese Adressen haben das Präfix FE80::/64. Die letzten 64 Bits der Adresse werden Schnittstellen-ID genannt und von der 48-Bit langen MAC-Adresse abgeleitet. Mit dieser Adresse kann diese Schnittstelle direkt mit benachbarten Hosts, wie zum Beispiel einem IP-Router, kommunizieren und so ohne manuelle Konfiguration weitere IP-Parameter per „Neighbor Discovery Protocol“ bekommen. DHCP ist somit nicht mehr für eine automatische Konfiguration notwendig.

Abbildung 20: Link-local-Adresse

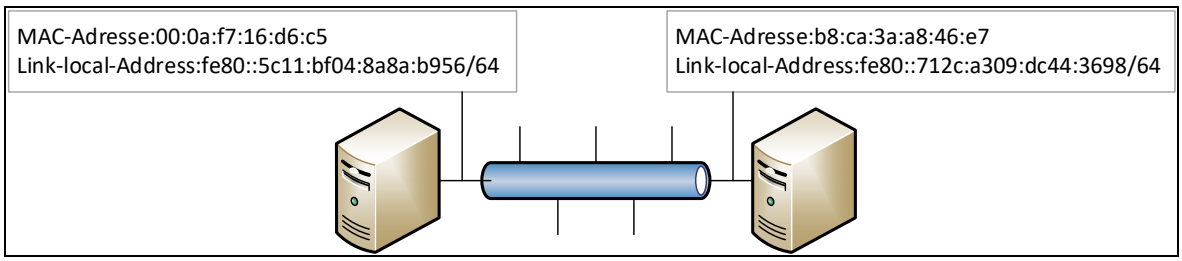


Abbildung 21: IPv6-Netzwerk mit verbindungslokalen Adressen

Verbindungslokale Adressen werden nur innerhalb geschlossener Netzwerksegmente eingesetzt. Möchte man ein Heim-, oder Firmennetz mit mehreren Subnetzen aufbauen, müssen „Unique-Local-Unicast“-Adressen verwendet werden. Diese entsprechen den privaten Adressen, die aus IPv4 bekannt sind.

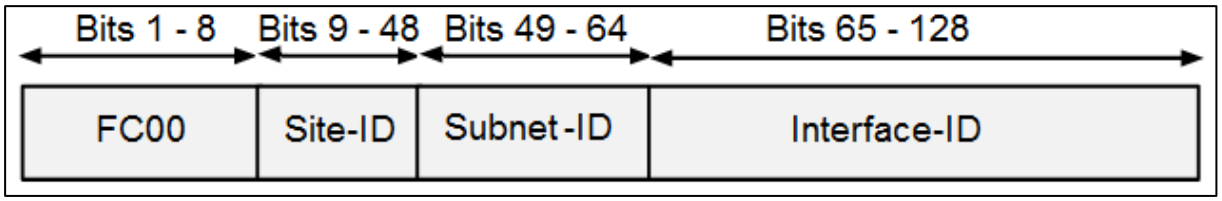


Abbildung 22: Unique-local-Adresse

Beispiel: FC01:2345:6789:0123:0123:4567:89AB:CDEF

- Der Präfix FC gibt an, dass es sich um eine Unique-local-Adresse handelt.
- Die Site-ID soll in Zukunft global vergebene Unique-local-Adressen eindeutig kennzeichnen. Zurzeit kann die Site-ID beliebige Werte annehmen (z.B. nur Nullen).
- Die Subnet-ID gibt mit 16 Bit wie bei IPv4 das Subnetz an.
- Mit der Interface-ID wird jeder Netzwerkcontroller im Subnetz eindeutig identifiziert.

Die Netz-ID lautet somit FC01:2345:6789:0123::/64.

1.2.6 Wichtige Protokolle von Layer 2 bis Layer 4

1.2.6.1 IP (Internet Protocol)

Das Internet Protocol der Version 4 und 6 läuft auf Schicht 3 des OSI-Modells ab (Network Layer) und dient zum Transport von Daten. Die grundlegenden Dateneinheiten werden als IP-Datagramme bezeichnet. Das IP arbeitet verbindungslos. Es wird keine Garantie dafür übernommen, dass die gesendeten Daten auch wirklich beim Empfänger ankommen.

1.2.6.2 ICMP (Internet Control Message Protocol)

Das Internet Control Message Protocol ist ebenfalls auf dem Network Layer angesiedelt. Es liefert Routing-Informationen und unterstützt IP bei Bearbeitung von Fehlermeldungen. Die ICMP-Message ist im Datenteil eines IP-Datagramms enthalten. Mögliche Fehlermeldungen bzw. Informationen sind:

- host unreachable (Zielrechner ist nicht erreichbar)
- route redirect (bessere Route zum Zielrechner gefunden)
- time exceeded (Lebenszeit eines Datagramms ist abgelaufen)
- incorrect datagram header (modifizierter Kopf des Datagramms)
- Übertragungszeit von Daten ist mittels Zeitsynchronisation feststellbar („ping“)

1.2.6.3 ARP (Address Resolution Protocol)

Das „Address Resolution Protocol“ ist für die Umsetzung von IP-Adressen in Hardware-MAC-Adressen zuständig. Es stellt also die Verbindung zwischen Layer 2 und 3 her. Dies ist insbesondere deshalb wichtig, da beispielsweise Switches nur Layer 2 „sprechen“, aber mit IP-Adressen nichts anfangen können. ARP ermöglicht also den Einsatz von Switches in Netzen, deren Datenaustausch rein auf IP beruht. MAC-Adressen sind weltweit eindeutig; IP-Adressen

können in nicht zusammenhängenden Netzen mehrfach genutzt werden.

1.2.6.4 TCP und UDP

TCP (Transmission Control Protocol) und UDP (User Datagram Protocol) führen Datenaustausch zwischen Programmen auf der Basis von IP durch. Sie arbeiten auf Schicht 4 des OSI-Modells. TCP ist ein verbindungsorientiertes und zustandsabhängiges Protokoll („stateful“). Vor dem Senden von Daten wird zunächst eine bidirektionale Verbindung zwischen Sender und Empfänger aufgenommen („handshake“). Verlorengegangene Pakete werden erneut gesendet. Anwendungen, bei denen die fehlerlose Übertragung von Informationen wichtiger ist als eine maximale Übertragungsgeschwindigkeit oder möglichst geringe Laufzeit, verwenden dieses Protokoll (z.B. telnet, ftp).
UDP arbeitet verbindungslos („stateless“). Eine Folge davon ist, dass die Übertragung der Daten ohne Zustellgarantie erfolgt. Die Daten können dadurch mit geringerer Verzögerung beim Empfänger ausgewertet werden, da dieser nicht erst überprüfen muss, ob ein Paket im Datenstrom fehlte. Dies ist für Echtzeitanwendungen, z.B. bei Internettelefonie oder Netzwerk-/Onlinespielen sehr relevant. Die korrekte Übertragung muss dann von einer übergeordneten Schicht sichergestellt werden.

2 Versuchsvorbereitung

2.1 Kurzfragen

Warum ist es mit einem Hub möglich, fremden Datenverkehr im LAN „mitzuhören“, jedoch nicht mit einem Switch? Gemeint ist hier ein *unmanaged* Switch ohne spezielle Monitoring Anschlüsse.

Warum steht an einem Switch in der Regel an allen Ports die volle Geschwindigkeit zur Verfügung, an einem Hub jedoch nicht?

Warum wurde IPv6 entwickelt und welche Vorteile gegenüber IPv4 bringt es mit sich? Nennen Sie mindestens 4 Vorteile.

Notieren Sie folgende IPv6 Adresse so kurz wie möglich. Achten Sie darauf, dass die Adresse eindeutig bleibt. Adresse: FC00:0000:041E:0001:0000:0000:04FF:C1A3

Einem Unternehmen wird das IP-Subnetz 213.55.77.160/28 zugewiesen.
Sie wollen innerhalb dieses Unternehmensnetzes den Host 213.55.77.176 erreichen. Können die IP-Pakete direkt zugestellt werden oder ist dies nur über einen IP-Router möglich? Begründen Sie Ihre Antwort durch eine kurze Rechnung.

Berechnen Sie die Broadcast-Adresse des obigen IP-Subnetzes. Wie gehen Sie hierbei vor?

2.2 Berechnungen in IPv6-Netzen

Ein kleines Unternehmen möchte ein IPv6-Testnetzwerk bereitstellen. Das Netz soll aus zwei Subnetzen bestehen und vereinfacht folgenden Aufbau haben (siehe Abbildung 23).

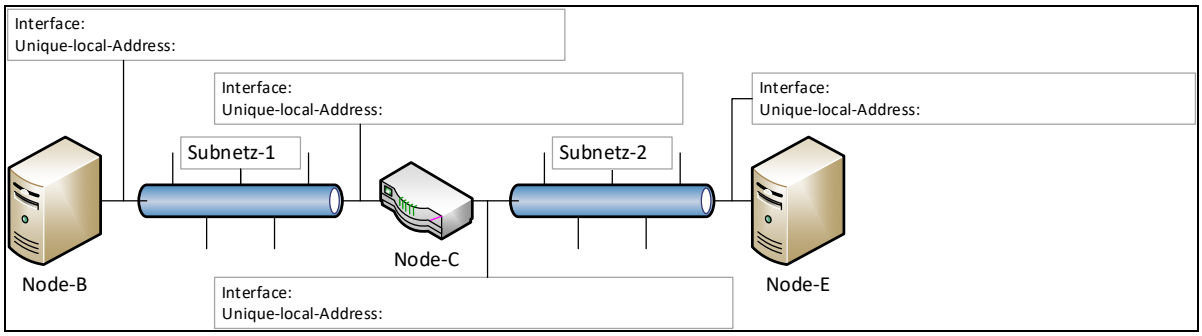


Abbildung 23: Versuchsaufbau (Vorbereitung)

Geben Sie für alle Schnittstellen eine gültige Unique-local-Unicast-Adresse an. Halten Sie die Adressen so einfach wie möglich und vergessen sie nicht das Präfix zu kennzeichnen. Beachten Sie, dass ein Router in seiner Rolle als Verbindungsstück zwischen zwei (oder mehr) Subnetzen auch zwei (oder mehr) IP-Adressen besitzt (Sie können die Adressen direkt in die obige Abbildung eintragen).

Mit welchen Adressen lässt sich von Node-B aus, Node-E mit einem ping-Befehl erreichen?

_____

_____

_____

2.3 Router im Netzwerk

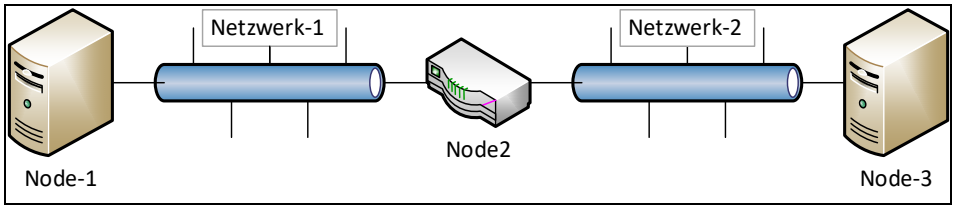


Abbildung 24: Versuchsaufbau (Vorbereitung)

a) Welche Funktionen erfüllt der Router „Node-2“ in Abbildung 24, wenn Netzwerk-1 und -2 jeweils eigenständige IP-Subnetze sind? Was passiert, wenn der Router durch einen Switch ersetzt wird?

_____

_____

_____

_____

b) Können Sie ausgehend von Node-1 mit einem IP-Broadcast an 255.255.255.255 Node-3 erreichen? Begründen Sie Ihre Antwort!

_____

_____

_____

_____

2.4 Netzwerke mit Subnetzen

In Abbildung 25 ist ein IPv4-Netzwerk mit drei Subnetzen dargestellt. Bitte entwerfen Sie eine mögliche Netzaufteilung auf OSI-Schicht 3 für die drei IPv4-Teilnetze. Verwenden sie für alle Subnetze jeweils ein eigenständiges, getrenntes IPv4-Subnetz mit CIDR-Notation **/16 (Class B Netz)**. Geben Sie für alle Geräte die zugehörigen IPv4-Adressen an und notieren Sie, wo nötig, die Subnetzmasken. Beachten Sie, dass ein Router in seiner Rolle als Verbindungsstück zwischen zwei (oder mehr) Subnetzen auch zwei (oder mehr) IP-Adressen besitzt.

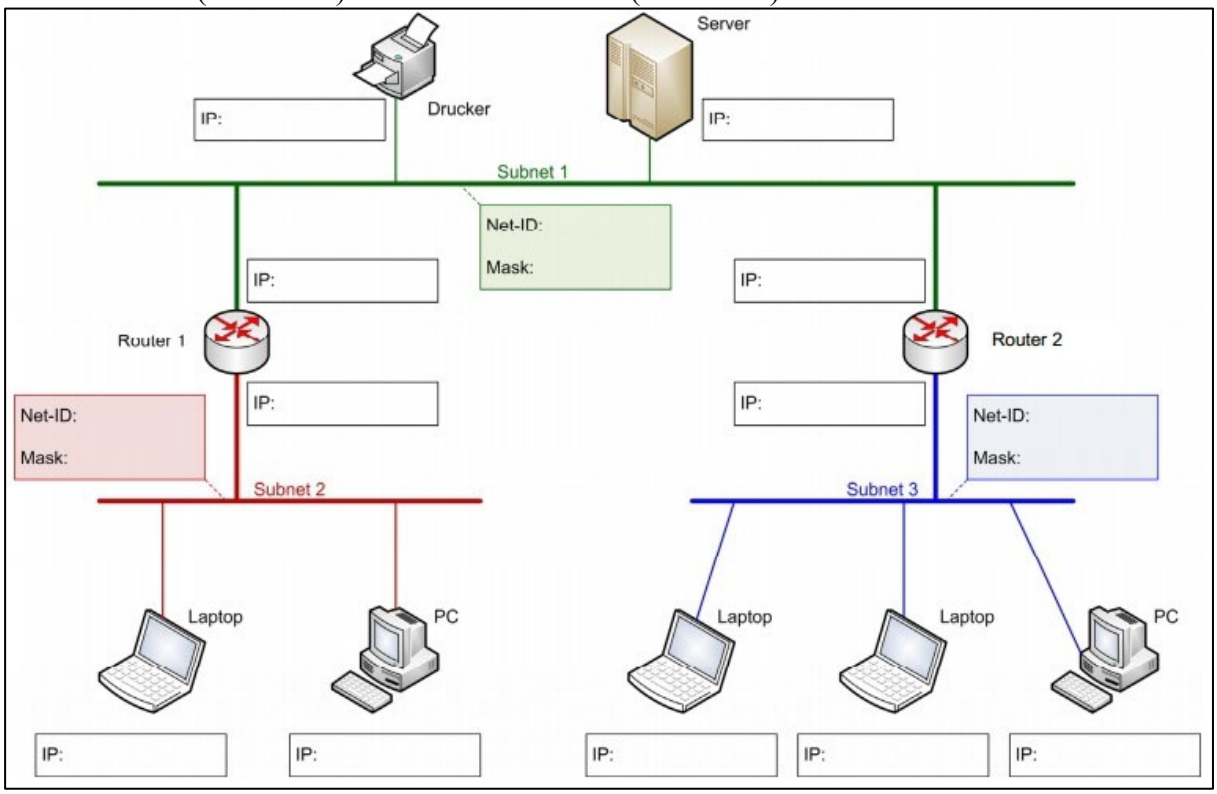


Abbildung 25: IPv4-Netzwerk mit Subnetzen (Vorbereitung)

2.5 Weitere Vorbereitung

Sie erhalten bei Bedarf zu Beginn Ihres Versuches eine kurze Einführung in die Bedienung der Systeme. Alle für die erfolgreiche Durchführung des Versuchs notwendigen Kommandos finden Sie auch im Anhang (bitte bereiten Sie sich entsprechend vor). Es wird erwartet, dass Sie die Vergabe einer IP-Adresse, eines Subnetzes und einer Route selbstständig vornehmen können.

3 Versuchsdurchführung

Hinweis: Bei der Durchführung kommen 5 Rechner zum Einsatz. Auf Node A läuft Windows 7. Auf Node B/C/D/E läuft Ubuntu 15.10. Die Netzwerkverbindungen unter Windows 7 wurden aus Gründen der Übersichtlichkeit umbenannt zu „eth0“ und „eth1“.

3.1 Ermittlung der Einstellungen in der Ausgangssituation

Melden Sie sich an den einzelnen Rechnern an. Die Zugangsdaten erhalten Sie von Ihrem Laborbetreuer. Melden sie sich in der Konsole von Ubuntu mit „sudo –s“ an, um alle benötigten Rechte zu erhalten.
Lesen Sie die ARP-Tabelle von einem Rechner Ihrer Wahl aus und notieren Sie das Ergebnis. Was fällt Ihnen auf? Erläutern Sie Ihr Ergebnis.

_____
_____
_____
_____

3.2 Verbinden zweier Netze mit einem IPv4-Router

Realisieren Sie gemäß Abbildung 26 das dargestellt Netzwerk (zunächst ohne Node-B). Verkabeln Sie die Rechner und vergeben Sie im Anschluss alle IP-Adressen und IP-Routen. Nutzen Sie für Ubuntu die Befehle nach 4.3 und gehen Sie für Windows nach 4.4 vor. Verwenden Sie eine von Ihnen entworfene Konfiguration (IPv4-Adressen, Subnetze, Einträge in den Routingtabellen usw.) und notieren Sie diese in Abbildung 26 sowie in Tabelle 6. Testen Sie die Funktion der internen Loopback-Interfaces (127.0.0.1) sowie alle Rechnerverbindungen mit dem Befehl ping. Dokumentieren Sie die Testergebnisse in Tabelle 7.

Falls Sie mit einem Rechner keine Kommunikationsverbindung herstellen können, versuchen Sie den Fehler zu beheben bzw. einzugrenzen. Prüfen Sie dazu alle Kabelverbindungen, Interface-Einstellungen sowie Einträge in den Routing- und ARP-Tabellen.

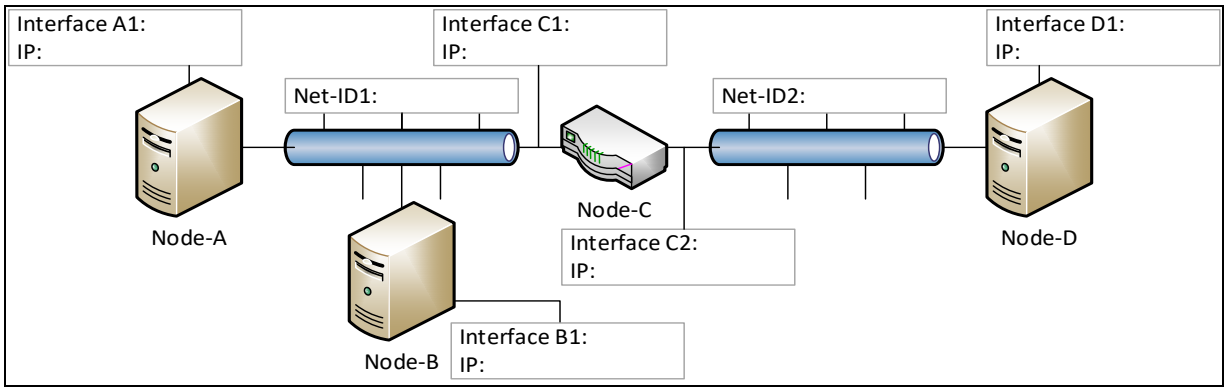


Abbildung 26: Versuchsaufbau 3.2

Bei der Notation von MAC-Adressen und Link-Lokalen IPv6 Adressen reichen die letzten 4 Zeichen.

Tabelle 6: Netzwerkkonfiguration für Versuchsaufbau 3.2

	Node-A	**Node-C**	**Node-D**
Loopback-Interface	lo0	lo0	lo0
IPv4-Adresse	127.0.0.1	127.0.0.1	127.0.0.1
1. Interface			
MAC-Adresse	: : : : :	: : : : :	: : : : :
IPv4-Adresse	. . .	. . .	. . .
Broadcast-Adresse	. . .	. . .	. . .
Netzmaske	. . .	. . .	. . .
2. Interface			
MAC-Adresse	: : : : :	: : : : :	: : : : :
IPv4-Adresse	. . .	. . .	. . .
Broadcast-Adresse	. . .	. . .	. . .
Netzmaske	. . .	. . .	. . .

Überprüfen Sie mit einem ICMP-Request (Ping) alle Kommunikationswege. Die Zuordnung von Namen und Node kann in der Datei „hosts“ (exemplarisch auf Node-A, Verknüpfung auf dem Desktop) festgelegt werden (siehe Kapitel 4.2). Namen können frei gewählt werden.

Tabelle 7: Überprüfung der Kommunikationswege für Versuchsaufbau 3.2

Absender		Empfänger			Kommunikation über Name		IP	
	IP-Adresse		Name	IP-Adresse	ja	nein	ja	nein
Node-A	loopback/localhost	Node-A	lo0	127.0.0.1				
Node-A	. . .	Node-C		. . .				
Node-A	. . .	Node-C		. . .				
Node-A	. . .	Node-D		. . .				
Node-C	loopback/localhost	Node-C	lo0	127.0.0.1				
Node-C	. . .	Node-A		. . .				
Node-C	. . .	Node-D		. . .				
Node-D	loopback/localhost	Node-D	lo0	127.0.0.1				
Node-D	. . .	Node-C		. . .				
Node-D	. . .	Node-C		. . .				
Node-D	. . .	Node-A		. . .				

3.3 Aktualisierung von ARP-Tabellen

Integrieren Sie jetzt Node-B (Abbildung 26) in Ihr Netzwerk (Netzwerkkabel einstecken und Rechner konfigurieren). Kommunizieren Sie noch nicht von Node-B aus mit anderen Rechnern des Netzes.

Notieren Sie den Inhalt der ARP-Tabelle von Router C in Tabelle 8. Die letzten 4 Zeichen reichen bei den MAC-Adressen aus.

Tabelle 8: ARP-Tabelle von Node-C vor der Kommunikation

Device	IP-Adresse	MAC-Adresse
	. . .	: : : : :
	. . .	: : : : :
	. . .	: : : : :
	. . .	: : : : :
	. . .	: : : : :
	. . .	: : : : :

Benutzen Sie jetzt den Befehl ping, um von Node-B ausgehend mit den anderen Rechnern zu kommunizieren. Sind Änderungen in der ARP-Tabelle von Node-C festzustellen? Notieren Sie Ihr Ergebnis in Tabelle 9. Füllen Sie nur neue Einträge vollständig aus.

Tabelle 9: ARP-Tabelle von Node-C nach der Kommunikation

Node	IP-Adresse	MAC-Adresse	Neuer Eintrag?
	. . .	: : : : :	
	. . .	: : : : :	
	. . .	: : : : :	
	. . .	: : : : :	
	. . .	: : : : :	
	. . .	: : : : :	

Können Sie ausgehend von Node-D die beiden anderen Rechner (Node-A und Node-B) gleichzeitig mit einem einzigen Aufruf erreichen? Begründen Sie Ihre Antwort.

_____
_____
_____

Ziehen Sie nun jegliche Steckverbindungen und entfernen Sie die Namenseinträge in den Dateien „hosts“.

3.4 Autokonfiguration bei IPv6

Stellen Sie sicher, dass alle Steckverbindungen gezogen sind. Nehmen Sie keine weiteren Konfigurationen vor und lassen Sie sich auf Node-A die Interfacekonfiguration anzeigen (ipconfig –all). Stecken Sie nun eine Steckverbindung von Node-A in den Hub. Lesen Sie anschließend in der Konsole erneut die Interfacekonfiguration der gesteckten Verbindung aus. Was fällt Ihnen auf?

_____
_____
_____
_____

Realisieren Sie nun das in Abbildung 27 dargestellte Netzwerk und tragen Sie die gewünschten Daten ein. Für Node-B muss nichts eingetragen werden. Die letzten 4 Zeichen reichen bei den verbindungslokalen Adressen aus.

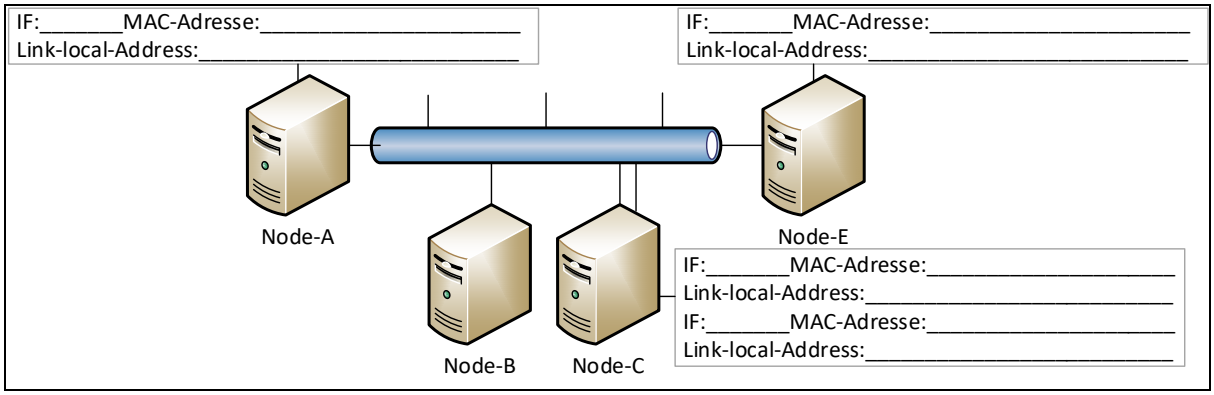


Abbildung 27: Versuchsaufbau 3.4.1

Überprüfen Sie die Funktionsfähigkeit einer beliebigen notierten Rechnerverbindungen mit dem Befehl ping bzw. ping6. Falls der Platz für die Adressen in Abbildung 27 zu gering ist, notieren Sie sie auf dem Blattrand. Bei Bedarf können Sie den Netzwerkverkehr mit Wireshark auf Node-A mitschneiden.

3.5 Subnetze in IPv6

Im nächsten Abschnitt soll das autokonfigurierte IPv6-Netzwerk gemäß Abbildung 28 in zwei Subnetze unterteilt werden. Konfigurieren Sie die Netzwerkschnittstellen mit Unique-local-Unicast-Adressen gemäß der Vorbereitung 2.2 und tragen Sie die benötigten Routen ein (siehe Kapitel 4.3.2 und 4.3.5). Die Wahl des Interfaces auf Node-B und Node-E bleibt Ihnen überlassen

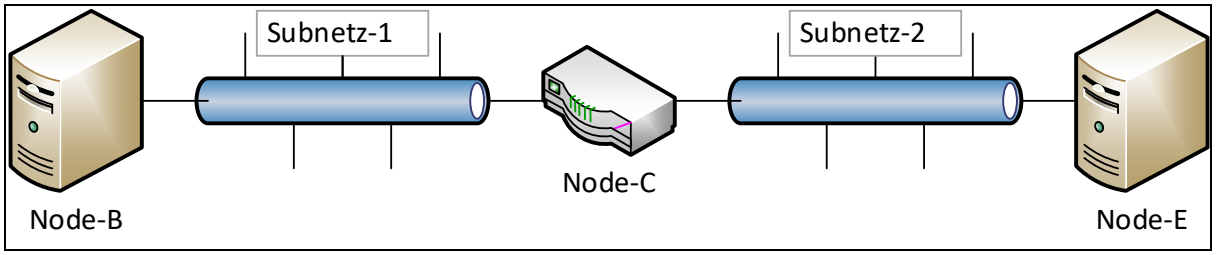


Abbildung 28: Versuchsaufbau 3.5

Testen Sie die Verbindung zwischen Node-B und Node-E mit dem ping6-Befehl und den Unique-Local-Unicast-Adressen und notieren Sie die Interfaceparameter in Tabelle 10.

Tabelle 10: Netzwerkkonfiguration von Versuchsaufbau 3.5

	Node-B	**Node-E**
Loopback-Interface	lo0	lo0
IPv6-Adresse	::1	::1
1. Interface		
MAC-Adresse	: : : : :	siehe Abb. 27
verbindungslokale-Adresse		siehe Abb. 27
Unique-Local-Unicast-Adresse		

Unter welchen Bedingungen kann man von dem ausschließlich autokonfigurierten Node-A aus die anderen Nodes erreichen? Sofern es eine Möglichkeit gibt, wie kann man von Node-A die Unique-Local-Unicast-Adressen von Node-E erreichen?

_____
_____
_____
_____
_____

3.6 Alternatives Routing

Stellen Sie sicher, dass alle Steckverbindungen gezogen sind. Stellen Sie sicher, dass auf dem Windows-PC IPv4 aktiviert ist.

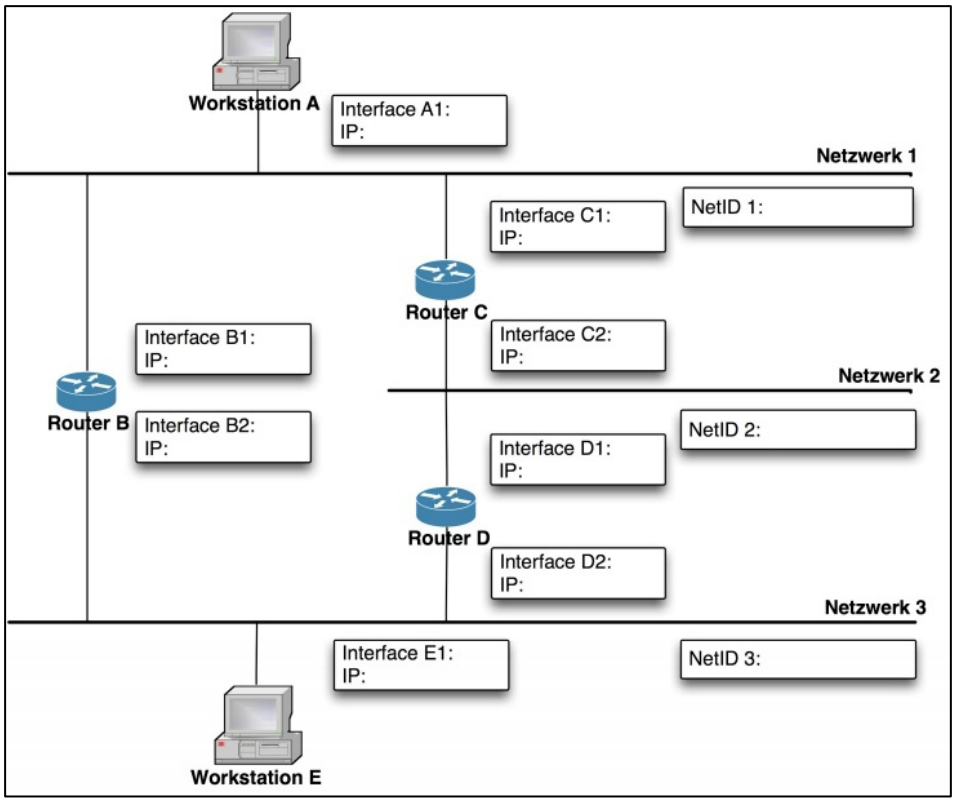


Abbildung 29: Versuchsaufbau 3.6

Realisieren Sie den Versuchsaufbau entsprechend Abbildung 29. Überprüfen Sie alle Verbindungen (dem Laborbetreuer vorführen). Zeigen Sie, welche Änderungen Sie vornehmen müssen, damit der Weg zwischen Node-A und Node-E einmal über Node-B und einmal über Node-C/-D verläuft.
Falls Sie einen der Rechner nicht erreichen können, analysieren Sie selbstständig den Fehler. Denken Sie daran, dass Sie stets eine Route für den Hin-und Rückweg benötigen, damit das Netz voll funktionsfähig ist.

Nutzen Sie den Befehl tracert und tragen Sie die Ergebnisse in Tabelle 12 und Tabelle 13 ein.

Tabelle 11: Routenverfolgung von Node-A nach Node-E

Hop-Nr.	IP-Adresse
1	. . .
2	. . .
3	. . .
4	. . .
5	. . .
6	. . .

Tabelle 12: Routenverfolgung von Node-A nach Node-E (Alternative)

Hop-Nr.	IP-Adresse
1	. . .
2	. . .
3	. . .
4	. . .
5	. . .
6	. . .

Ist es möglich, ausgehend von Node-A, Node-E zur gleichen Zeit und ohne Änderung der Konfiguration über beide möglichen Wege zu erreichen (Routenverfolgung zum Überprüfen nutzen)?

_____
_____
_____
_____
_____
_____

Fahren Sie die Nodes B/C/D/E herunter. Löschen Sie bevor Sie Node-A herunterfahren alle manuellen IP-Netzwerkkonfigurationen sowie eventuell noch vorhandene Einträge in der hosts-Datei.

4 Anhang

Die nachfolgenden Seiten enthalten einige für die erfolgreiche Durchführung des Laborversuches notwendige Informationen. Detailliertere Angaben finden Sie in der einschlägigen Literatur.

4.1 Notwendige Programme und Dateien

Für diesen Versuch benötigen Sie einige netzwerkspezifische Dateien und ein Tool für die kommandozeilenorientierte Befehlseingabe. Für die Eingabe von Befehlen liegt auf dem Windwos-basierten Node auf dem Desktop eine Verknüpfung mit dem Programm ‚cmd' welches als Administrator geöffnet werden muss. Unter Ubuntu kann das Terminal, welches unter anderem in der Schnellstartleiste abgelegt ist genutzt werden. Eine Befehlsausführung kann jederzeit mit der Tastenkombination Strg+C abgebrochen werden. An Dateien ist Für Sie im Moment nur die Datei hosts aus dem Verzeichnis /etc bei Ubuntu von Bedeutung. Für Windows ist eine Verknüpfung dieser Datei auf dem Desktop angelegt, die mit dem Texteditor bearbeitet werden kann. Abbildung 30 stellt die Dateistruktur von Linux stark vereinfacht dar.

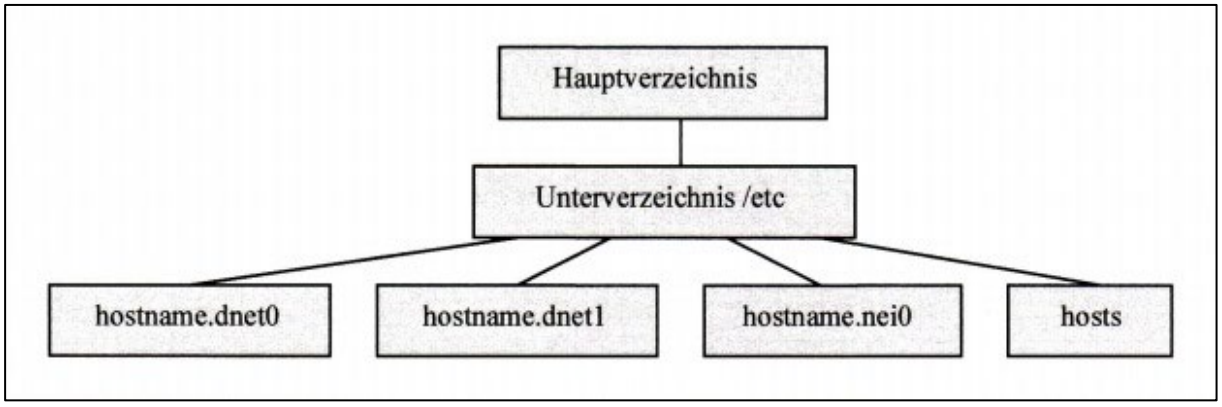


Abbildung 30: Beispiel für Dateistrukturen

4.2 Die Datei hosts

Diese Datei enthält die Zuordnung von IP-Adressen zu Interface-Namen. In sehr kleinen Netzen ist eine manuelle Bearbeitung der „hosts"-Dateien noch möglich. In größeren Netzen wird die Datei durch spezielle Programme (routing daemons) automatisch aktualisiert oder DNS-Server (Domain Name Service) sind für diese Zuordnung zuständig. Im Laborversuch müssen Sie diese Datei bei Bedarf von Hand aktualisieren!

Beispiel: Wenn Sie z.B. eine Nachricht an das Netzwerkgerät „Odin" senden wollen, können Sie entweder die IP-Adresse 172.16.1.1 oder den Namen „Odin" angeben. Wenn Sie den Namen benutzen, muss Ihr Rechner die Bezeichnung „Odin" in die IP-Adresse 172.16.1.1 umsetzen können. Die benötigten Informationen befinden sich in der Datei „hosts". Wenn nicht, müssen diese Daten eingefügt werden. Benutzen Sie hierfür das Programm „nano". Für Windows ist eine Verknüpfung dieser Datei auf dem Desktop angelegt, die mit dem Texteditor bearbeitet werden kann.

```
#
# Internet host table
#
127.0.0.1       localhost
192.168.1.1     Thor
172.16.1.1      Odin
```

Abbildung 31: Beispiel für die Datei "hosts"

4.3 Relevante Befehle für den Laborversuch

Für die Durchführung des Versuches benötigen Sie die Kommandos netstat, ifconfig, ipconfig, arp, ping, ping6, route, traceroute und tracert. Wenn Sie einen Befehl ohne Argument eingeben, wird oft die Syntax auf dem Bildschirm angezeigt. Ausführlichere Informationen erhalten Sie durch Eingabe des Kommandos man <Befehlsname>.

Tabelle 13: Relevante Befehle der Versuchsdurchführung

Tool	Beschreibung
arp	Zuordnung von MAC- und IP-Adressen
ifconfig & ipconfig	Initialisierung, Konfiguration und Analyse der Network-Interfaces
netstat	Status des Netzes, Adressen usw.
ping & ping6	Überprüfung von Netzverbindungen zwischen Rechnern mit ICMP
ip route	Manuelle Manipulation der Routingtabelle
traceroute & tracert	Zeigt die einzelnen Routingstationen (Hops) bis zum Zielrechner an

Nachfolgend sind diese Kommandos mit einigen für die Arbeit im Labor wesentlichen Parametern aufgeführt.

4.3.1 Befehl arp

Mit diesem Befehl kann der ARP-Cache dargestellt und auch modifiziert werden.

Tabelle 14: arp-Befehl

Kommando	Erklärung
arp -a	Alle Einträge des ARP-Caches anzeigen
arp -d	Einträge löschen
arp -s	Einträge hinzufügen

```
# arp –a
Net to Media Table

  Device        IP-Address          Mask          Flags      Phys Addr
-------------------------------------------------------------------------
 dnet0         224.0.0.1       255.255.255.255              01:00:5e:00:00:01
 dnet1         224.0.0.1       255.255.255.255              01:00:5e:00:00:01
 dnet1         Merlin          255.255.255.255    SP        00:00:92:9b:37:50
 dnet0         Leo             255.255.255.255    SP        00:00:d1:1b:bb:65
 dnet1         224.0.0.0       240.0.0.0          SM        01:00:5e:00:00:00
 dnet0         224.0.0.0       240.0.0.0          SM        01:00:5e:00:00:00
```

Abbildung 32: Beispiel für eine ARP-Tabelle

4.3.2 Befehl ifconfig und ipconfig

Mit diesem Befehl ist die Initialisierung, Steuerung und Überprüfung der Netzschnittstelle (Netzwerkkarte) möglich. Sie erhalten z. B. Informationen über die Internet-Adresse, die Netzmaske und die Broadcast-Adresse des Interfaces. Alle in den Statusinformationen genannten Einstellungen können modifiziert werden.

Tabelle 15: ifconfig-Befehl für Ubuntu

Kommando	Erklärung
ifconfig	Anzeige der aktuellen Einstellungen aller Interfaces, inklusive MAC-Adresse, IP-Adresse, Subnetz-Maske, Net-ID und Broadcast-Adresse
ifconfig <Schnittstellenname>	Anzeige der Einstellungen des Interfaces <Schnittstellenname> . Die Schnittstellen haben unter Linux in der Regel Bezeichnungen wie eth0 , eth1 , eth2 , usw.

Tabelle 16: ipconfig-Befehl für Windows

Kommando	Erklärung
ipconfig	Anzeige der aktuellen Einstellungen aller Interfaces, inklusive IP-Adresse, Subnetz-Maske, Net-ID und Broadcast-Adresse
ipconfig -all	Anzeige aller aktuellen Einstellungen aller Interfaces, inklusive der MAC-Adresse

Beispiel: Konfiguration des Interfaces „eth0“ mit IPv4-Adresse 195.80.16.45 und Subnetz Maske 255.255.255.0 unter Ubuntu:

ifconfig eth0 195.80.16.45/24

oder alternativ: **ifconfig eth0 195.80.16.45 netmask 255.255.255.0**

Der Befehl erlaubt auch das An- und Abschalten des Interfaces unter Ubuntu. Ein abgeschaltetes Interface taucht nicht in der Auflistung der aktuellen Einstellungen aller Interfaces auf.

ifconfig eth0 up (Einschalten)

ifconfig eth0 down (Ausschalten)

Um eine IPv6-Adresse unter Ubuntu zu vergeben kann folgender Befehl verwendet werden:

ifconfig eth0 inet6 add fd00:0:0:1::1/64

Um IPv6-Adressen von einem Interface zu entfernen kann folgender Befehl verwendet werden:

ifconfig eth0 inet6 del fd00:0:0:1::1/64

4.3.3 Befehl netstat

Alle vorhandenen Interfaces können mit diesem Befehl angezeigt werden. Mögliche Optionen:

- Namen der Interfaces feststellen
- MAC-Adressen der Netzkarten ermitteln
- Statistiken über gesendete/empfangene Pakete und die Anzahl aufgetretener Fehler ausgeben
- Anzahl der Kollisionen nennen

Tabelle 17: netstat-Befehl

Kommando	Erklärung
netstat –a	alle Verbindungen anzeigen
netstat –e	Layer 2 - Statistik
netstat –i	Interface-Informationen anzeigen
netstat –p	Protokoll (TCP/UDP)
netstat –r	Routing-Tabelle
netstat –s	Statistik (ausführlich)
netstat –interval [sec]	automatische Updates

4.3.4 Befehl ping und ping6

Mit dem Befehl ping wird die Erreichbarkeit eines Rechners getestet.

Tabelle 18: ping-Befehl unter Ubuntu

Kommando	Erläuterung
ping <host>	Sendet Pings unter IPv4 zum angegebenen Host
ping6 <host>	Sendet Pings unter IPv6 zum angegebenen Host
Ping6 –I <Interface> <link-local-address>	Zur Nutzung mit Verbindungslokalen Adressen
ping –c <count> <host>	Anzahl zu sendender Pings

Tabelle 19: ping-Befehl unter Windows

Kommando	Erläuterung
ping <host>	Sendet IPv4 oder IPv6 Pings zum angegebenen Host
ping <host> -n <count>	Anzahl zu sendender Pings

Hinweis:
Die Befehlsausführung kann jederzeit mit der Tastenkombination Strg+C abgebrochen werden.

4.3.5 Befehl ip route und route

Der lokale Rechner benötigt Routing-Informationen, um eine Netzverbindung zu einem Rechner außerhalb seines eigenen lokalen Subnetzes aufzubauen. Diese Informationen können manuell oder über route daemons (automatisch ablaufende Prozesse) gewonnen werden. Bei beiden Verfahren wird die Routing-Tabelle des Kerns modifiziert. Die manuelle Manipulation der Routing-Tabelle erfolgt mit dem Kommando ip route. Folgende Optionen stehen dem Anwender zur Verfügung:

Tabelle 20: route-Befehl für Ubuntu

Kommando	Erklärung
ip route	Gibt die Routing-Tabelle aus
ip route add <Netzwerk-ID> via <Gateway>	Fügt eine neue Route hinzu

ip route del <Netzwerk-ID>	Entfernt eine IPv4-Route
ip -6 route add <Netzwerk-ID> via <Gateway>	Fügt eine neue IPv6-Route hinzu
ip -6 route del <Netzwerk-ID>	Entfernt eine IPv6-Route

Hinweis:
Die dynamische Aktualisierung der Routing-Tabelle mit Hilfe von route daemons oder gate daemons ist nur in einem komplexen Netz sinnvoll, weil dabei System- und Netzressourcen verbraucht werden. Im Labor werden die Tabelleneinträge manuell vorgenommen.
Beispiel für das Hinzufügen einer neuen Route unter IPv4 bei Ubuntu:

ip route add 192.168.1.0/24 via 192.168.2.1

Hiermit wird einem Rechner mitgeteilt, dass er das fremde Netz 192.168.1.0/24 über den IP-Router 192.168.2.1 erreichen kann. Man beachte, dass hinter add eine Netzwerk-ID steht (Angabe immer mit Subnetz, hier mit /24 geschehen) und hinter via nur eine IP-Adresse. Die IP-Adresse des Gateways muss selbstverständlich im selben IP-Netz liegen wie der Host-PC. Anstatt add kann mit del dieser Routing Eintrag wieder entfernt werden.

ip route del 192.168.1.0/24

4.3.6 Befehl traceroute und tracert

Hinweis: Unter Windows lautet der Befehl „tracert“, unter Ubuntu „traceroute“.

Dieser Befehl ermöglicht es, in einem gerouteten Netzwerk die einzelnen Stationen (Hops) bis zu einem Zielhost anzeigen zu lassen. So können Sie auch kontrollieren, ob Ihre Konfiguration korrekt ist oder ob das Paket an einer bestimmt Stelle auf seinem gerouteten Weg hängen bleibt. Des Weiteren zeigt traceroute die Verzögerungszeiten zu jedem einzelnen Hop. Hier sehen Sie ein Beispiel für eine Routenverfolgung (unter Windows) aus dem Ostfalia Netz zum Host „heise.de“ im Internet:

```
Routenverfolgung zu heise.de [193.99.144.80]  über maximal 30 Abschnitte:
1     <1 ms     <1 ms     <1 ms   141.41.40.1
2     1 ms      <1 ms      <1 ms   WiN-IP-Gate.FH-Wolfenbuettel.DE [141.41.1.2]
3     1 ms      1 ms       1 ms    xr-bra1-ge8-4.x-win.dfn.de [188.1.233.41]
4     3 ms      3 ms       3 ms    xr-mag1-te1-1.x-win.dfn.de [188.1.144.245]
5     5 ms      4 ms       5 ms    xr-pot1-te2-1.x-win.dfn.de [188.1.144.253]
6     5 ms      5 ms       5 ms    zr-pot1-te0-0-0-0.x-win.dfn.de [188.1.145.162]
7     18 ms  17 ms  17 ms  te3-1.c302.f.de.plusline.net [80.81.193.132]
8     17 ms  17 ms  17 ms  82.98.98.102
9     18 ms  17 ms  18 ms  redirector.heise.de [193.99.144.80]
Ablaufverfolgung beendet.
```

Sie sehen, dass die Pakete zu „heise.de“ über insgesamt acht IP-Router weitergeleitet werden. Hop 9 ist der Zielhost selbst.

Häufig können Sie sogar anhand der Router-Namen erkennen, welchen geographischen Weg ein IP-Paket in etwa nimmt. Dies ist immer dann möglich, wenn die Administratoren der Router diese mit Hinweisen über deren Standort versehen (häufig die Stadt).

In unserem Fall beginnt das Paket seinen Weg zunächst am lokalen Router 141.41.40.1, wird dann über den Hauptrouter der Ostfalia (Hop 2) in das Deutsche Forschungsnetz (DFN) übergeben. Mit den Kürzeln bra1, mag1 und pot1 sind vermutlich die Städtenamen

Braunschweig, Magdeburg und Potsdam gemeint, in denen die Router 3 bis 6 aufgestellt sind. Man beachte insbesondere, dass das Paket lediglich 5 ms von Wolfenbüttel bis Potsdam benötigt, was ein hervorragender Wert ist. Hinter Hop 6 wird das Paket in das Netz von Plusline übergeben und gelangt nach einem weiteren Hop (vermutlich innerhalb des Rechenzentrums) schließlich zum Ziel.

4.4 Netzwerkkonfiguration unter Windows

Um unter Windows die Netzwerkschnittstellen zu konfigurieren muss zunächst die grafische Oberfläche des Adapters durch einen Doppelklick auf „eth0“ bzw. „eth1“ geöffnet werden. Anschließend kann durch einen Klick auf „Eigenschaften“ die Übersicht über die vorhandenen Protokolle geöffnet werden. Durch die Checkboxen vor der Protokollbezeichnung kann das Protokoll (de)aktiviert werden.

Um eine einfache IPv4-Konfiguration zu erstellen muss ein Doppelklick auf „Internetprotokoll Version 4“ ausgeführt werden. Dort können die Parameter unter „Folgende IP-Adresse verwenden“ eingetragen werden.

Dasselbe gilt für die IPv6-Konfiguration, die unter „Internetprotokoll Version 6“ zu erreichen ist.

Um unter Windows Routingeinträge vorzunehmen kann das vorhandene Feld „Default Gateway“ verwendet werden.

Hinweis: Für den vollen Funktionsumfang unter Windows muss die Konsole mit Administratorrechten gestartet werden. Rechtsklicken Sie hierfür auf „cmd“ und wählen Sie „Als Administrator ausführen“.

4.5 Beispiele für Routingtabellen eines Subnetzes

Diese Tabellen können in der Praxis auch anders aufgebaut sein und andere Einträge enthalten. Oft findet sich in Routing-Tabellen ein Default-Eintrag (0.0.0.0, sinngemäß mit der Bedeutung „alle anderen“), damit Datenpakete für unbekannte Zielrechner ebenfalls weitergeleitet werden können.

Routing-Tabelle für R0:	IN	Mask	Gateway	Device
	195.80.16.0	255.255.255.0	195.80.16.1	B
	195.80.17.0	255.255.255.128	195.80.16.11	B
	195.80.17.128	255.255.255.128	195.80.16.21	B
	0.0.0.0	0.0.0.0	(übergeordneter Router)	
Routing-Tabelle für R1:	**IN**	**Mask**	**Gateway**	**Device**
	195.80.16.0	255.255.255.0	195.80.16.11	A
	195.80.17.128	255.255.255.128	195.80.16.21	A
	195.80.17.0	255.255.255.128	195.80.17.1	B
	0.0.0.0	0.0.0.0	195.80.16.1	A
Routing-Tabelle für R2:	**IN**	**Mask**	**Gateway**	**Device**
	195.80.16.0	255.255.255.0	195.80.16.21	A
	195.80.17.128	255.255.255.128	195.80.17.129	B
	195.80.17.0	255.255.255.128	195.80.16.11	A
	0.0.0.0	0.0.0.0	195.80.16.1	A
Routing-Tabelle für Host 1:	**IN**	**Mask**	**Gateway**	**Device**
	195.80.16.0	255.255.255.0	195.80.16.2	A
	0.0.0.0	0.0.0.0	195.80.16.1	A

Abbildung 33: Beispiel für eine Routingtabelle

Sollen z. B. Daten aus dem Subnetz IN 1 zu einer Zieladresse in IN 2 übertragen werden, überprüft der Router R1 die Einträge in der Routing-Tabelle. Dort findet er den entsprechenden Eintrag (IP 195.80.17.128) und erfährt, dass die Daten über IP 195.80.16.21 geleitet werden müssen.