Netzwerktopologie

Topologie	Verbindungen	Point-to-Point	Konnekt.
Stern	$n-1$	möglich	1
Bus	$n-1$	nein	1
Baum	$n-1$	eingeschränkt	1
Ring	n	nein	2
Voll	$n(n-1)$	möglich	viele

Multiplexing

Auswahl des nächsten Hops bei großen Netzwerken

Fluten Sende das Paket an alle Nachbarn
Hot Potato Routing Sende an einen zufälligen Nachbarn
Routingtabellen In jedem Switch mit einem Eintrag pro Ziel. Enthält Info über kürzeste Wege

Serviceprimitive

Request (Req) Anfrage an Layer eines Service auzuführen
Indication (Ind) Layer zeigt seinem Nutzer, dass etwas passiert ist (asynchrone Benachrichtigung)
Response (Res) Nutzer von höherem Layer beantwortet Indication
Confirmation (Conf) Der ursprüngliche Dienstaufrufer wird über die Beendigung des Servicerequests informiert

Korrektheitsanforderung

Completeness Alle gesendeten Nachrichten werden irgendwann zugestellt
Correctness Alle Daten die ankommen, sind auch genau die, die losgeschickt wurden (unverändert, keine Bitfehler)
Reihenfolgegetreu Nachrichten und Bytesequenzen kommen in der korrekten Reihenfolge an
Verlässlich Sicher, Verfügbar, ...
Bestätigt Erhalt von Daten wird dem Sender bestätigt

Verbindungsorientiert

Verbindungsorientierte Dienste müssen Primitive bereitstellen um Verbindungen handhaben zu können

CONNECT Einrichtung der Verbindung
LISTEN Warten auf Verbindungsanfragen
INCOMING_CONN Anzeige eingehender Con.Requests
ACCEPT Annahme einer Verbindung
DISCONNECT Terminierung einer Verbindung

Layering

Vorteile
- Komplexität verwalten & beherrschen
- Änderung der Implementierung transparent
- Ideales Netzwerk

Nachteile
- Funktionen vl redundant
- selbe Information für verschiedene Layer nötig
- Layer n benötigt eventuell Einblick in Layern $n+x$

Informationsgehalt nach Shannon

Anforderungen an Funktion für Informationsgehalt eines Zeichens

- $I(x) = -Id(P(x)) = ld(\frac{1}{P(x)})$
- $I(x)$ reziprok zu $P(x)$
- $(P(x) = 1) \Rightarrow (I(x) = 0)$
- $I(x, y) = I(x) + I(y)$

Entropie

Die Entropie bezeichnet den durchschnittlichen Informationsgehalt einer Zeichenquelle.

Eingangsentropie $H(X) = \sum_{i=1}^{n} P(x_i) * ld(\frac{1}{P(x_i)})$
Verbundentropie $H(X,Y) = \sum_{i=1}^{n} \sum_{j=1}^{m} P(x_i, y_i) * ld(\frac{1}{P(x_i, y_i)})$
Verlustentropie $H(X|Y) = \sum_{i=1}^{n} \sum_{j=1}^{m} P(x_i, y_i) * ld(\frac{1}{P(x_i, y_i)}) = H(X,Y) - H(Y)$
Störentropie $H(Y|X) = \sum_{i=1}^{n} \sum_{j=1}^{m} P(y_i, x_i) * ld(\frac{1}{P(y_i, x_i)}) = H(X,Y) - H(X)$
Transinformation $H_T = H(X) + H(Y) - H(X,Y)$

Architekturvoraussetzungen

für das Internet

Generalität Unterstütze alle möglichen Sets von Applikationen
Heterogenität Verbinde alle Arten von Netzwerktechnologien
Robustheit Wichtiger als Effizienz
Erweiterbarkeit Wichtiger als Effizienz
Skalierbarkeit Spätere Entdeckung

Medium Access Control (MAC)

Annahmen für dynamische Kanalzuweisung

Einkanalannahme Nur ein Kanal für alle Stationen und für alle Nachrichten
Kollisionsannahme Nur je ein Frame zeitgleich fehlerfrei übertragbar
Stationsmodell
- N unabhängige Stationen
- Mögliches Lastmodell: Wahrscheinlichkeit des Generierens eines Pakets im Intervall t ist x*T, mit x konstant

Zeitmodell
- Kontinuierlich: Übertragungen können jederzeit stattfinden
- Geslottet: Zeit ist in Slots eingeteilt, Übertragung kann nur an Slotgrenzen beginnen

Carrier Sensing (CSMA)
- Stationen können (oder nicht) erkennen, ob der Kanal frei oder in Benutzung ist
- Falls Kanal als belegt angesehen, so wird nichts übertragen

Carrier Sensing CSMA

Höre bevor du redest, sende nichts, wenn Medium belegt
1-Persistent CSMA Falls belegt, so warte bis frei und sende dann → Probleme entstehen, wenn mehrere nach der jetzigen Nachricht senden wollen
Non-Persistent CSMA Wenn Kanal frei so übertrage, wenn Kanal belegt, so warte eine zufällige Zeit vor dem nächsten Freiheitstest
P-Persistent CSMA Kombiniert bisherige Ideen + geslottete Zeit, Warte ständig auf freiwerden des Kanals übertrage aber nicht sofort

Collision Detetion - CSMA/CD

Abhängig vom physischen Layer können Kollisionen erkannt werden, so warte eine zufällige Zeit k

Bit-Map-Protokoll

Stationen melden Sendewunsch während eines Reservierungsslots an

- Verhalten bei geringer Last: Wenn kaum ein Paket versendet werden soll, so wiederholt das Medium die Contentionslots → Wartezeit
- Verhalten bei großer Last: Hoher und stabiler Durchsatz mit vernachlässigbarem Overhead
- Bit-Map ist ein Carrier Sense Protokoll

Limited Contention Protokoll

Idee 1
- Anpassen der Stationsanzahl per Contentionslot
- Contentionslots sind gut für den Durchsatz, bei geringer Last können wir es uns aber nicht leisten, auf die Antworten zu warten → Stationen müssen sich dynamisch einen Slot teilen

Idee 2 Adaptives Baumprotokoll := Verwende verschiedene Auflösungslevel für die Wettbewerbsslots

Ethernetversionen

Switched Ethernet mehrere Stationen über ein Kabel
Fast Ethernet wie Switched nur mit 10ns Bitzeit
Gigabit Ethernet jedes Kabel hat genau zwei Maschinen angehängt

mit Switch
- Keine geteilten Kollisionsdomänen, benötigen kein CSMA-CD
- Fullduplexoperation auf jedem Link

mit Hub
- Kollisionen, Halbduples, CSMA-CD
- Maximale Kabellänge 25 Meter

Quellenkodierung

ordnet jedem Zeichen einen binären Code zu.
Kodierung/Dekodierung möglichst einfach und mittlerer Kodierungsaufwand möglichst klein. Versucht möglichst alle Redundanzen zu vermeiden.

Code	Stellen	Bemerkung
Block	fest	keine Berücksichtigung der Informationsgehalte
Fano	variabel	• Sortieren von groß nach klein • Teilen in ähnlich große Partition • Vordere Gruppe erhält 0 • Hintere Gruppe erhält 1
Huffmann	variabel	• Zusammenfassen der beiden kleinsten Auftrittswahrscheinlichkeiten • Die kleinere Auftrittswahrscheinlichkeit erhält eine 1 als Codierungsziffer

Kanalcodierung

Kanalcodierung fügt Redundanz ein, um Übertragungfehler zu detektieren.
Dies kann man durch das Anhängen eines Paritätsbits erreichen oder Cyclic Redundancy Check. Hierbei werden die Daten als Binärzahl aufgefasst. Die Zahl wird um n Nullen erweitert und durch eine festgelegte Zahl der Länge $n+1$ dividiert. Die Division wird ohne Brücksichtigung des Übertrages durchgeführt, also modulo 2. Der Rest, der bei der Division entsteht, wird anschließend auf die Binärzahl aufaddiert. Wird die berechnete Zahl fehlerfrei übertragen, entsteht bei einer Division durch die Prüfzahl kein Rest. Erhält der Empfänger bei der Division einen Rest, weiß er, dass die Daten fehlerhaft übertragen worden sind.

Leitungskodierung

Die Leitungkodierung bildet das Codealphabet auf physikalische Signale ab. Der binäre Datenstrom, der von Quellenkodierung und Kanalkodierung erzeugt wird, muss also durch verschiedene analoge Signalmuster dargestellt werden.

Digital-Analog-Wandlung

Zur Digital-Analog-Wandlung können die folgenden Modulationsverfahren genutzt werden

- Amplitudenmodulation
- Phasenmodulation
- Frequenzmodulation

Digitale Signalcodes

Ein digitaler Signalcode heißt selbsttaktend, wenn aus dem Datensignal der Übertragungstakt gewonnen werden kann. Dies hat den Vorteil, dass man keine weitere Leitung zur Übertragung des Taktes braucht, um Sender und Empfänger synchron zu halten. Ein Code heißt gleichstromfrei, wenn die Summe der Impulse Null ergibt.

NRZ none return to zero
- 1 bei positiver Pegel
- 0 bei neutrale Pegel

RZ Return to Zero
- 1 in Folge eines positiven und neutralen Impulses
- 0 nach zwei Neutralen

Biphase-L Manchestercode
- 1 in Folge negativ und positiv Impuls
- 0 in Folge positiv, negativ
- gleichstromfrei, selbsttaktend

Differential Manchestercode
- 0 durch Pegelwechsel am Anfang
- 1 durch Fehlen eines Pegelwechsels
- in Mitte des Bitzeitintervall stets Pegelwechsel
- gleichstromfrei, selbsttaktend, verpolungssicher

AMI
- Ternärcode
- 0 bei neutralem Pegel
- 1 alternierend durch negativen/positiven Pegel
- gleichstromfrei, verpolungssicher

Scrambler

Scrambler erhöhen die Anzahl der Pegelwechsel in einem nicht selbstaktendem Code, damit trotzdem ausreichend Synchroninformation im Datenstrom enthalten ist. Dies wird durch Division durch ein Binärpolynom erreicht. Das Ergebnis der Division wird übertragen.

Analog-Digital-Wandlung

zeit- und wertkontinuierliche Analogsignal in ein zeit- und wertdiskretes Difgitalsignal umwandeln

Quantisierung

Zur Wandlung in ein Digitalsignal wird das Analogsignal in Zeitabständen δt abgetastet. Der ermittelte Wert wird quantisiert. Die Differenz zwischen Quantisierungstufe und echtem Wert wird auch Quantisierungsfehler genannt.

Abtasttheorem

die Abtastrate f_A muss größer als die doppelte Maximalfrequenz f_{Smax} sein: $f_A \geq 2 * f_{Smax}$

Kanalkapazität

Die Kanalkapazität K_{Rausch} hängt von der Signalleistung P_S und der Rauschleistung P_R im Medium aber

Kapazität $K_{Rausch} = f_{Kmax} * ld(\frac{P_s}{P_R} + 1)$

falls rauschfrei $K = 2f_{Kmax} * ld(n)$

Rauschabstand $r = 10 * lg(\frac{P_S}{P_R})$

Kap. bei Rauschabstand $K_{Rausch} \approx \frac{f_{Kmax} * r}{3}$

Fehlerbehandlung

Stop-and-Wait

- Sender sendet erst, wenn er Quittung für Empfang des letzten Paketes erhalten hat
- Rahmennummerierung benötigt

Automatic Repeat reQuest Verfahren

Mit ARQ bezeichnet man Verfahren, die mit einer Kombination von Fehlererkennung, Zeitgebern, Bestätigung und Übertragungswiederholungen arbeiten.

Go-Back-N

- Sender benötigt Puffer, um noch nicht bestätigte Pakete zwischenzuspeichern
- Falls eine Bestätigung für ein Paket ausbleibt, sendet er alle Pakete ab diesem erneut
- hohe Belastung der Übertragungswege

Selective reject ARQ

- wie Go-Back-N
- allerdings wird nur das fehlerhafte Paket neuübertragen
- Empfänger benötigt Puffer, um ausgefallene Pakete nachträglich einfügen zu können

Kollisionsbehandlung

CSMA/CD

Falls der Kanal frei ist, beginnt er zu senden. Nun horcht er (eine festgelegte Zeit lang), ob es Kollisionen gibt. In dieser Zeit erreicht das Signal jeden am Kanal angeschlossenen Sender, sofern das Netz Standardgerecht verlegt ist. Stellt der Sender eine Kollision fest z.B. in Form überlagerter Signale, zieht er sich vom Kanal zurück und versucht später wieder den Sendevorgang zu wiederholen. Gab es aber keine Überlagerung kann er ohne zu horchen weitersenden, da kein anderer Sender mit dem Sendevorgang beginnt, wenn der Kanal belegt ist.

Token Verfahren

In einem logischen Ring wird ein Token im Kreis durchgereicht. Der der den Token hat darf senden. Er wandelt den Freitoken in den Header eines Datenpaketes um und sendet nun fortlaufend Daten. Der Empfänger nimmt die Daten nicht vom Ring, sondern schickt sie mit gesetztem Bestätigungsbit weiter. So erhält der Sender gleich noch eine Bestätigung.

DQDB-Zellen-Verfahren

Auf zwei entgengengesetzten Bussen werden von den Endsystemen laufend Zellen gesendet. Möchte eine Station in eine Richtung senden, setzt sie in einem Fenster der entgegengesetzten Richtung das Busy-Bit. Alle Stationen, an denen dieses Paket vorbeiläuft, erhöhen ihren Warteschlagencounter. Wenn sie nun auch senden wollen, wissen sie mittels dieses Counters an welcher Stelle der Warteschlange sie stehen. Wenn auf der gefragten Leitung nun eine leere Zelle vorbeiläuft, senken alle Stationen in der Warteschlange ihre Position um eins und die erste Station beginnt zu senden.

Multiplexing

Techniken, die es ermöglichen mehrere Verbindungen auf einem Kanal zu halten

Raummultiplex räumlichen Zuweisung von Kanälen an Regionen. Unter Beachtung eines Mindestabstandes kann ein Kanal doppelt genutzt werden

Zeitmultiplex Aufteilung eines Kanals auf mehrere Teilnehmer mittels einer synchronen oder asynchronen Zeitscheibentechnik (konstante oder wechselnde Zeitabstände)

Frequenzmultiplex Aufteilung eines Kanals auf mehrere Teilnehmer mittels Zuweisung verschiedener Frequenzbänder an verschiedene Teilnehmer

Codemultiplex Aufteilung eines Kanals auf mehrere Teilnehmer mittels Zuweisung verschiedener Kodierungen an verschiedene Teilnehmer

Wartezeiten

Wartesystem Aufträge werden in eine endlose Warteschlange eingereiht.

Verlustsystem Aufträge, die nicht direkt bearbeitet werden können, gehen verloren.

Warteverlustsystem Aufträge werden in eine endliche Warteschlange eingereiht. Ist die Warteschlange voll, gehen weitere Aufträge verloren.

Verteilung der Ankunftsabstände

TA... Abstand der ankommenden Aufträge
F_{TA} ist exponentialverteilt: $F_{TA}(t) = 1 - e^{-\lambda t} = P(TA = \delta t)$
Dichtefunktion: $f_{TA} = \lambda e^{\lambda t}$

Verteilung der Anzahl der Ankünfte

K... Anzahl der Ankünfte in einem Zeitintervall T
$P(K = k) = \frac{\lambda^k * T^k}{k!} e^{\lambda T}$

Little'sches Gesetz

TA mittlere Verweilzeit eines Auftrages. Die Verweilzeit eines Auftrages setzt sich aus Wartezeit und Bearbeitungszeit des Auftrages zusammen.
N Anzahl der Aufträge; λ Ankunftsrate der Aufträge
$N = \lambda * T \leftrightarrow T = \frac{N}{\lambda}$
Das Gesetz ist unabhängig von der Wahl der Bedienstrategie und sogar von der Größe des Warteschlangennetzes.

Internetworking

Pfaderkennung - Selbstlernen

- jeder Switch hat eine Switchtabelle
- Eintrag: (MAC-Adresse, Interface, Zeitstempel)
- beim Empfang eines Frames lernt der Switch den Ort des Senders kennen (Rückwärtslernen)

Weiterleiten

- falls Ziel bekannt so prüfe, ob es in das selbe Segment gehört aus dem es kommt $\rightarrow$ verwerfen,
- sonst leite es passend weiter
- andernfalls flute das Netzwerk damit

Rückwärtslernen in Bridges - Bootstrapping

- Flute, falls nicht bekannt wohin gesendet werden muss
- verwerfe wenn bekannt, dass es nicht nötig ist, oder
- leite spezifisch weiter, wenn das Ziel bekannt ist

Gateways

Wenn selbst Router nicht ausreichend, dann sind Higher-Layer-Verbindungen notwendig. Arbeit auf dem Transportlevel und oberhalb, zum Beispiel für Transcodierung

Verbindung einzelner LANs

- Physisches Layer - Repeater und Hub
- Data-Link-Layer - Bridges und Switches
- Netzwerklayer - Routing
- Higher-Layer - Gateways

Netzwerklayer

Durchsuchen der Routingtabelle

- Suche nach übereinstimmender Hostadresse (Flag H)
- Suche dann nach passender Netzwerkadresse
- Drittens, Suche nach einem Defaulteintrag

Switching Fabric

Switching mittels Speicher
- Herkömmliche Rechner mit Switching unter direkter CPU-Kontrolle
- Kopieren der Pakete in den Systemspeicher
- Geschwindigkeit limitiert durch die Speicherbandbreite

Switching mittels BUS
- Übertragung von Datagrammen intern über einen Bus
- Switchinggeschwindikeit limitiert durch die Busbandbreite
- typ. 1Gbps Bus, ausreichend für Heim und Businessrouter

Switching mittels Verbindungsnetzwerk (Crossbar)
- Überwinden der Bandbreitenbeschränkungen von Busen
- Design: Fragmentierung von Datagrammen in Zellen fester Größe, wobei nun die Zellen durch das Fabric geswitcht werden
- Bis zu 1.28 Tbps Switchinggeschwindigkeit

IP Paketformat

Version Versionsnummer des eingesetzten IP
IHL IP Header Length in 32 Bit Worten
Typ des Dienstes: Infos zur Priorisierung
Totale Länge Die gesamtlänge in Bytes inklusive Header
Identifier Wenn Fragmentierung auftritt, bekommt jedes zugehörige Paket den selben Identifier
Flags DF (don't fragment), MF (more fragments, alle außer das letzte Paket haben dies gesetzt)
Fragment Offset Position des Fragments im ursprünglichen Paket
TTL Zähler für die Hopanzahl, wird an jedem Router dekrementiert, sobald gleich 0 → verwerfen
Protokoll Spezifiziert verwendetes Protokoll
Headerchecksum Erlaubt Verifizierung der Inhalte im IP Header
Quell und Zieladressen identifizieren der Quelle und des Ziels
Optionen bis 40 Byte, zur Erweiterung verwendet

Klassen von IP-Adressen

Class A rießige Organisationen, bis 16 Mil. Hosts
Class B große Organisationen, bis 65 Tausend Hosts
Class C kleine Organisationen, bis 255 Hosts
Class D Multicast, keine Netzwerk/Host Hierarchie
Class E reserviert
Loopback 127.xxx.xxx.xxx ist zum Testen reserviert, hierauf versendete Pakete werden als eingehende behandelt
Broadcast alles 1en

IP-Adressierung

- IPv4 Adresse: 32 Bit Identifier für Hosts oder Routinginterfaces
- Interface: Verbindung zwischen Host und dem physischen Link. IP Adressen werden an das jeweilige Interface vergeben

CIDR: Classless Inter Domain Routing

- Überwinden der Klassengrenzen durch Supernetting
- ISPs können nun Class C Blocks zu einem großen Block zusammenfassen
- 'Longest match routing' auf maskierten Adressen
- Beispiel: Alle in Europa vergebenen Adressen teilen sich einen gemeinsamen Prefix → nur ein Eintrag für alle Verbindungen nach Europa in den meisten amerikanischen Routern

NAT - Network Address Translation

- Lokale Netzwerke haben nur eine der Außenwelt bekannte IP-Adresse, somit hat nicht jedes Gerät eine vom ISP bereitgestellte Adresse
- Möglichkeit intern Adressen zu vergeben ohne die Außenwelt informieren zu müssen
- Wechsel des ISPs möglich, ohne intern Adressen zu verändern
- Geräte im Netzwerk nicht von außen ansprechbar (Sicherheitsfaktor)
- 16 Bit Portnummernfeld → 60'000 simultane Verbindung mit nur einer einzigen LAN-Side Adresse

ICMP: Internet Control Message Protocol

- Verwendet von Hosts und Routern um auf Netzwerkebene Informationen auszutauschen
- In Netzwerkebenen oberhalb von IP werden ICMP Nachrichten als IP Datagramme versendet
- ICMP Nachrichten: Typ, Code + erste 8 Bytes des den Fehler auslösenden IP-Datagramms

IPv6

- Header mit 40 Byte Größe (20 Byte mehr als bei IPv4 mit 32 Bit Adressen)
- Fragmentierung ist nicht mehr erlaubt
- Headerformat hilft bei schneller Verarbeitung und Weiterleitung
- Checksummen → komplett entfernt
- Optionen → Erlaubt, aber außerhalb des Headers
- ICMPv6 → Zusätzliche Nachrichtentypen + Multicastgruppenmanagementfunktionen

IPv6 Header

Priority Signalisiert die Priotität der Datagramme im Fluss
Flow Label Identifiziert Datagramme im selben Fluss
Next Header Identifiziert Layer der höheren Schicht für Daten

Routing Algorithmen

- Ein Router führt einen Routingalgorithmus aus, um zu entscheiden, an welchem Ausgang ein eingehendes Paket weiter übertragen werden sollte.

  Verbindungsorientiert nur beim Verbindungsaufbau
  Verbindungslos entweder für jedes Paket oder periodisch ausgeführt

- Oftmals unter Verwendung von Metriken → Zuweisung eines Kostenfaktors an jeden Link, bspw. Anzahl an Hops, Kosten eines Links,...

Nichtadaptive Routingalgorithmen

keine Rücksicht auf aktuellen Netzwerkzustand
Fluten jedes eingehende Paket wird auf jede ausgehende Linie geschickt, außer auf die Herkunftslinie
Zufallsrouting Jedes ankommende Paket wird auf einen zufälligen Ausgang geschickt, außer auf den Quellausgang → es bahnt sich seinen Weg sozusagen durch den Router

Adaptive Routingalgorithmen

Berücksichtigen den aktuellen Netzwerkzustand
Zentralisiertes adaptives Routing Anpassen an die vorherrschende Verkehrslast; Ein Routingkontrollcenter muss ins Netzwerk eingebaut sein, welches periodisch den Linkstatus der Router erhält und kürzeste Routen berechnet und diese an die Router sendet
Isoliertes adaptives Routing benötigt keinen Informationsaustausch zwischen Routern; Routingentscheidungen werden nur anhand der Informationen des lokalen Routers getroffen, wie bei Hotpotato oder Rückwärtslernen
Verteiltes adaptives Routing Router tauschen periodisch Infos aus und aktualisieren Weiterleitungstabellen; Finde einen guten Pfad durch das Netzwerk, welcher einen von der Quelle zum Ziel führt; Graphabstraktion für Routingalgorithmen mit Linkkosten und Pfadkosten

Distanzvektorrouting Algorithmen

Iterativ Läuft bis keine Knoten mehr Informationen austauschen. Selbstterminierend → kein Stoppsignal
Asynchron Knoten müssen Informationen nicht getaktet austauschen
Verteilt Jeder Knoten kommuniziert nur mit seinem direkten Nachbarn
Distanztabellendatenstruktur Jeder Knoten hat seine eigene Spalte für jedes mögliche Ziel und Zeile für jeden direkt angeschlossenen Nachbarknoten

Vergleich zwischen Link-State und Distanzvektoralgorithmen

- Nachrichtenkomplexität

  LS mit N Knoten und E Links werden $O(n-e)$ Nachrichten versandt
  DV Austausch nur zwischen Nachbarn

- Konvergenzgeschwindigkeit

  LS $O(n^2)$ Algorithmus benötigt $O(N-E)$ Nachrichten (teils mit Oszillation)
  DV Konvergenzzeit variiert (Routingschleifen, Count to Infinity Problem, Oszillation)

- Robustheit (im Falle eines Routerausfalls)

  LS Ein Knoten kann falsche Linkkosten ausgeben; Jeder Knoten berechnet nur seine eigene Tabelle
  DV DV Knoten kann falsche Gewichte ausgeben; Jede Tabelle wird nun noch von anderen Routern verwendet → Fehler breiten sich über das ganze Netzwerk aus

Routing im Internet - Autonome Systeme

Das globale Internet besteht aus miteinander verbundenen AS

Stub AS kleine Unternehmen (ein Link zum Internet)
Multihomed AS große Unternehmen (mehrere Links, ohne Transitverkehr)
Transit AS Netzbetreiber

Zwei Level Routing

Intra-AS Administrator verantwortlich für die Auswahl (RIP, OSPF, IGRP)
Inter-AS Einheitlicher Standard (BGP)

Intra-AS und Inter-AS Routing

- Policy:
  - Inter AS: Admin möchte Kontrolle über sein Netz haben
  - Intra AS: ein einziger Admin, keine Policyentscheidungen nötig
- Skalierbarkeit: Hierarchisches Routing spart Tabellenplatz und sorgt für weniger Updateverkehr
- Performance:
  - Inter-AS: Policy wichtiger als Performance
  - Intra-AS: Performance als oberstes Gut

Transport Layer

Multiplexing und Demultiplexing

Hosts verwenden IP-Adressen und Portnummern um Segmente an korrekte Sockets zuzustellen

Multiplexing auf Sendeseite Sammeln von Daten an mehreren Sockets, verpacken der Daten mit Header zum Demultiplexing
Demultiplexing auf Empfangsseite Zustellen empfangener Segmente an den korrekten Socket

Verbindungslos (UDP) Erstelle Sockets mit Portnummern; Sockets werden übber Zweiertupel aus Ziel IP und Ziel Port identifiziert
Verbindungsorientiert (TCP) TCP Sockets werden durch ein Vierertupel aus Quell-IP, Quellport, ZielIP und Zielport identifiziert

verbindungsorientierte Kontrolle

Connect → Data → Disconnect

- T-Connect.Request(Zieladr., Quelladr)
- T-Connect.Indication(Zieladr., Quelladr.)
- T-Connect.Response(Antwortadresse)
- T-Connect.Confirmation(Antwortadresse)

CR (Connection Request) oder CC (Con. Confirm) TPDU

Drei Wege Handshake

- Verbindung wird Aufgabaut, sobald beide Verbindungsaufbau TPDUs bestätigt wurden
- Benötigt zusätzliches ACK (Acknowledgement) oder DT (Data)
- Packe hierzu eine Sequenznummer in die CR, ACK, CC, DATA TPDUs
- Muss durch die Gegenseite kopiert werden, und erlaubt den Verbindungsaufbau nur dann, wenn die korrekte Nummer bereit gestellt wird. Verwende Sequenznummern deshalb möglichst nicht schnell hintereinander erneut.

Verbindungsabbau

implizit Abbau der Netzwerklayerverbindung
explizit Verbindungsfreigabe mit Disconnect-TPDUs

Kann den Verlust von nicht bestätigten Daten nach sich ziehen. TCP verhindert dies, indem alle gesendeten PDUs vor Beenden der Verbindung bestätigt werden müssen.

Flusskontrolle

Pufferallokation

- Flusskontrolle abhängig von der Puffermöglichkeit
- Um ausstehdene Pakete zu unterstützen müssen diese entweder sofort und in korrekter Reihenfolge beim Empfänger ankommen oder es muss genügend Puffer vorhanden sein
- Empfänger verlangsamt den Sender oder Anforderung von Pufferspeicher durch den Sender
- Mitteilung des Empfängers an den Sender, dass nur noch so viel Puffer verfügbar ist (bei Sliding Window einfach das Sendefenster anpassen)

Continue und Stop

Einfachste Lösung: Sende Stopnachrichten wenn der Empfänger nicht schritthalten kann und Continue, sobald wieder Ressourcen vorhanden sind.
Beispiel: XON/XOFF: funktioniert aber nur bei Fullduplexverbindungen.

Implizite Flusskontrolle

Idee: Halte ACKs oder NACKs zurück, um den Sender zu verlangsamen, somit werden Fehlerkontrollmechanismen nun zur Flusskontrolle missbraucht werden.
Nachteil: Senderseitig keine Unterscheidung mehr möglich, ob Pakete verloren gingen, oder er verlangsamt werden soll, was in unnötigen Wiederholungsübertragungen resultiert.

Kreditbasierte Flusskontrolle

Der Empfänger gewährt dem Sender expliziten Kredit, sodass dieser meherere Pakete senden kann. Ist der Kredit aufgebraucht, so muss der Sender warten, bis er wieder neuen zugeteilt bekommt. Hierbei benötigen wir Fehlerkontrolle um auf verlorene Kreditnachrichten resultieren zu können

Permits und Acknowledgements

- Permits = Empfänger hat Pufferspeicher, sende also weiter
- Acknowledgements = Empfänger hat Anzahl X an Paketen empfangen
- Kombinierbar mit dynamisch wachsendem Pufferplatz beim Emfänger (Beispiel TCP)

Staukontrolle

Jedes Netzwerk kann nur eine gewisse Anzahl an Traffic pro Zeit transportieren, wenn nun mehr Traffic von den Quellen ausgeht, als das Netzwerk als nominelle Kapazität hat, so kommt es zu Staukollapsen und verlorenen Paketen. Immer $\lambda - in = \lambda - out$ (goodput)
Staukontrolle ist essentiell, um Schneeballeffekte zu vermeiden: Sobald ein Netzwerk einmal überladen ist, wird es Pakete verlieren. Nach Erkennung von Paketverlusten durch ein zuverlässiges Transportprotokoll, werden Pakete erneut übertragen, was die Last abermals erhöht

- Senderate jeder Quelle muss an die aktuelle Kapazität des Netzwerks angepasst werden
- Staukontrolle globales Problem, da abhängig von allen Routern, Weiterleitungsdisziplinen, Lastinjektionen usw ist.
- Flusskontrolle lokales Problem: die Quelle darf das Ziel nicht überlasten; nur Ziel und Quelle involviert

Design/Aktions Optionen

Open Loop Designe das System von Beginn an so, dass es korrekt funktioniert und man keine Korrekturen zur Laufzeit vornehmen muss
Closed Loop Verwende Feedback, um zu erlauben, dass sich der Sender an die Situation anpasst
Explizited Feedback die Stelle, an welcher der Stau auftritt informiert den Sender
Implizites Feedback der Sender extrahiert aus dem Netzwerkverhalten Informationen darüber, wie er sich verhalten sollte

- Erhöhen der Kapzität → teuer, kurzfristig nicht umsetzbar
- Reservierungen und Zugriffskontrolle - erlaube keinen zusätzlichen Verkehr wenn das Netzwerk stark ausgelastet ist → nur für schaltkreisbasierende Netzwerke verfügbar
- Reduzierung der Last in kleiner Granularität → Bringe einzelne Quellen dazu ihre Last zu reduzieren, sodass nichts terminiert werden muss (benötigt Feedback vom Netz: closed loop)
- Verwerfen von Paketen → Pufferplatz ist voll und alte Pakete werden verworfen. Für Medieninhalte sind neue wichtiger als alte Pakete

Choke Pakete

Sobald der Router einen Stau erkannt hat → sende Chokepakete. Chokepakete sagen dem Ziel, dass es seine Senderate verringern soll

Warnungsbits

Sobald ein Router feststellt, dass er von Stau betroffen ist, setzt er ein Warnbit in allen Paketen die er verschickt → da das Ziel das Warnungsbit in sein ACK Paket aufnimmt, erfährt die Quelle vom Stau und kann ihre Sendeleistung minimieren.

Random Early Detection

nutze verworfene Pakete als implizites Feedback, bereits bevor die Warteschlange voll ist, wirf also vorzeitig Pakete weg um Feedback zu geben. Mit steigender Staubelastung am Router kann die Entwurfswahrscheinlichkeit erhöht werden

TCP

Drei Wege Handshake

- Client sendet TCP SYN (SYN = 1, ACK = 0) an Server → spezifiziert initiale, nie benutzte Sequenznummer
- Server erhält SYN Paket und antwortet mit SYNACK (SYN = 1, ACK = 1) → Server alloziert Puffer und Spezifikation der initialen Sequenznummer des Servers
- Der Client erhält das SYNACK und antwortet hierauf mit einem ACK (SYN = 0, ACK = 1), hier können nun erstmals Daten enthalten sein

Terminieren einer Verbindung

- Client sendet ein TCP FIN
- Server empfängt das FIN, antwortet mit einem ACK und sendet ebenfalls ein FIN
- Client erhält ein FIN Segment, antwortet darauf mit ACK und geht in timed Wait Zustand, antwortet auf alle FINs mit ACKs
- Server erhält ein ACK, die Verbindung ist geschlossen

Sende- und Empfangspuffer

Sender Puffer um Fehlerkontrolle bereit zu stellen
Empfänger Zwischenspeichern von noch nicht abgerufenen oder nicht reihenfolgegetreu angekommenen Paketen

Flusskontrolle: Angebotenes Fenster

Der Empfänger kann seine Empfangpufferkapazitäten verkünden

Nagles Algor. - Selbsttaktung und Fenster

- TCP Selbsttaktung: Ankunft eines ACKs ist Zeichen, dass neue Daten auf das Netzwerk geschickt werden können
- falls sowohl angebotene Daten und das angebotene Fenster $>= MSS$ → sende ein volles Segment
- falls unbestätigte Daten auf dem Weg sind, so puffere neue Daten bis das MSS voll ist, andernfalls schicke die Daten sofort

Staukontrolle

- Implizites Feedback durch verworfene Pakete. Annahme: Stau als Hauptgrund für verworfene Pakete
- Fensterbasierte Staukontrolle: TCP führt Buch über die Anzahl an Bytes die es noch in das Netzwerk injezieren darf, diese Fenstergröße kann wachsen oder schrumpfen

AIMD - Sägezahnmuster der Last

- TCP verwendet AIMD, also *additive increase, multiplicative decrease* Taktik
- es wird kontinuierich auf zusätzliche Bandbreite geprüft und durch Erhöhung der Bandbreitengrenze wird das Netzwerk regelmäßig die multiplikative Verringerung ausführen → Sägezahnmuster

Application Layer

HTTP Statuscodes

200 OK: Anfrage okay, das angefragte Objekt folgt
301 Moved Permanently: das angefragte Objekt wurde verschoben, der neue Pfad folgt
400 Bad Request: Anfrage wurde nicht verstanden
404 Not Found: angefordertes Objekt konnte auf dem Server nicht gefunden werden
505 HTTP Version not supported

Cookies

- Cookieheaderzeile in der Antwort/Anfrage
- Cookiedatei wird auf dem Rechner des Hosts gespeichert und vom Browser verwaltet

- Speichern der Cookieinformationen in einer Backenddatenbank der Webseite

Webcaches (Proxyserver)

Bedienen der Clientanfrage ohne den urpsrünglichen Webserver dabei zu involvieren

- Nutzer stellt den Browser so ein, dass dieser über einen Cache auf das Netz zugreift
- Alle Anfragen des Browsers gehen zuerst an den Cache, hat er das angefragte Material, so wird er dieses an den Client schicken, oder andernfalls beim Webserver besorgen und dem Client dann weiterleiten
- Cache agiert als Client und Server
- Reduzieren von Antwortzeiten für Clientanfragen
- Reduzieren von Verkehr auf dem Zugangslink des ISPs
- Internet voller Caches erlaubt es armen Anbietern effektiv Inhalte zu übertragen

Webserver

Grundlegende Webserveraufgaben

- Zum Empfang von Anfragen bereitmachen
- Annehmen von Verbindungen und Anfragen
- Lesen und Verarbeiten von Anfragen
- Antworten auf Anfragen
- Bereitmachen und Annehmen von Anfragen

Prozessmodell

- Prozess werden alle benötigten Schritte zugewiesen, um eine Anfrage zu bearbeiten
- Wenn Bearbeitung abgeschlossen, so ist Prozess wieder in der Lage neue Verbindungen zu akzeptieren
- Typischerweise werden mehrere Prozesse benötigt
- falls ein Prozess blockiert, entscheidet das OS, welcher Prozess als nächstes ausgeführt werden darf
- Parallelität limitiert durch die Anzahl an Prozessen
- Vorteile: Synchronisation dem Prozessmodell inhärent; Absicherung zwischen Prozessen
- Nachteile: Langsam; Schwere Ausführbarkeit von Operationen, welche auf globalen Informationen beruhen

Threadmodell

- Verwende Threads anstelle von Prozessen
- Vorteile: Schneller als Prozesse; Teilen aktiv
- Nachteile: Benötigt OS Unterstützung; Kann per Prozess Limitierungen überlasten; Beschränkte Kontrolle über Schedulingentscheidungen

In-Kernel Modell

- ganzer Server im Kernel möglich
- meist nur statische Dateien vom Kernel bedient, andere Anfragen an regulären User-Space-Server
- Dedizierter Kernelthread für HTTP Anfragen
- Vorteile: Vermeidet Kopieren von und in den Userspace; Sehr schnell, solange eng in den Kernel integriert
- Nachteile: Bugs können OS crashen; Schwer zu debuggen/erweitern; Inhärent OS-spezifisch

Eventbasiertes Modell

- Verwenden eines einzelnen Webserverprozesses um mehrere Anfragen zu behandeln
- Vorteile: Sehr schnell, kein Kontextwechsel; Inhärentes Teilen ohne Locks; Komplette Kontrolle über die Schedulingentscheidungen; Kein komplexer OS-Support benötigt
- Nachteile: Per-Prozess Begrenzungen; Nicht jedes OS mit voll asynchroner E/A, so können beim Lesen immernoch Blockierungen entstehen; Flash verwendet immerhin Hilfsprozesse um dies zu verhindern

Mailzugriffsprotokolle

SMTP Zustellen/Speichern auf dem Empfangsserver
POP Post Office Protocol: Autorisierung und Download; POP3 ist zustandlos über mehrere Sitzungen
IMAP Internet Mail Access Protocol: Mehr Features aber komplexer; Behält alle Nachrichten am Server
HTTP Yahoo Mail, Hotmail, etc.

DNS - Domain Name System

verteilte Datenbank implementiert in der Hierarchie von vielen verschiedenen Nameservern Anwendungsschichtprotokoll für Hosts, Router und Nameserver zum Kommunizieren zur Namensauflösung

Sicherheit

Sicherheitsziele

Vertraulichkeit Verschickte oder gespeicherte Daten sollen nur einem bestimmten Nutzerkreis zugänglich sein; Vertraulichkeit von Instanzen wird auch als Anonymität bezeichnet
Datenintegrität Es sollte möglich sein, jede Veränderung von Daten zu erkennen, dies benötigt unter anderem, die Möglichkeit den Ersteller von Daten identifizieren zu können
Verantwortlichkeit Es sollte möglich sein, eine Instanz zu identifizieren, welche für irgendein Kommunikationsereignis zuständig ist
Verfügbarkeit Dienste sollten verfügbar sein und auch funktionieren
Kontrollierter Zugriff Nur autorisierte Instanzen solle in der Lage sein auf bestimmte Dienste oder Daten zuzugreifen

Bedrohnungen technisch definiert

Maskerade (Spoofing) Eine Instanz behauptet jemand Anderes zu sein
Abhören (Sniffing) Jemand versucht Daten zu lesen, welche er nicht lesen darf und soll
Autorisierungsverletzungen Eine Instanz verwendet Ressourcen die sie nicht verwenden darf
Verlust oder Veränderung von übertragener Information Veränderung oder Zerstörung von Daten
Fälschung von Daten Eine Instanz erzeugt Daten im Namen einer Anderen
Abstreiten von Kommunikationsereignissen Eine Instanz streitet seine Beteiligung an einem Kommunikationsereignis ab
Sabotage Jede Art von Aktion welche darauf abzielt, die Verfügbarkeit oder korrekte Funktion von Diensten zu reduzieren

Sicherheitsanalyse von gelayerten Protokollarchitekturen

Dimension 1: Auf welchem Interface findet der Angriff statt?
Dimension 2: Auf welchem Layer findet der Angriff statt?

Sicherheitsmechanismen

Physische Sicherheit Abschließen der Betriebsräume, Zutrittskontrolle; Schutz vor Überwachung der Umgebung
Personelle Sicherheit Sensitivität bei Mitarbeitern erzeugen; Überprüfung der Angestellten; Sicherheitstraining
Administrative Sicherheit Kontrollieren neuer Software; Prozeduren um Sicherheitsverstöße zu erkennen; Ansehen und Reagieren auf Audittrails
Ausstrahlungssicherheit Steuerung von Frequenzen und anderer elektromagnetischer Ausstrahlungen
Mediensicherheit Kontrollieren der Erstellung, Reproduktion und Zerstörung von Informationen; Scannen von Medien auf Schadsoftware
Lifecyclekontrollen Vertrauenswürdiges Systemdesign der Implementierung, Evaluation und Unterstüzung; Dokumentierung; Einhalten von Programmierstandards
Computersicherheit Schutz der Informationen, während diese auf Rechnern gespeichert oder verarbeitet werden; Schutz der Rechner selbst
Kommunikationssicherheit Schutz der Informationen beim Transport von einem zum anderen System; Schutz der Kommunikationsinfrastruktur an sich

Sicherheitsdienste

Authentisierung Grundlegender Sicherheitsdienst, welcher sicherstellt, dass eine Instanz tatsächlich die Identität hat, welche sie vorgibt zu haben
Integrität Kleiner Bruder der Authentisierung, da er sicherstellt, dass Daten, welche von einer gewissen Einheit erstellt worden sind, nicht ohne Erkennung verändert werden können
Vertraulichkeit Stellt sicher, dass die geschützen Daten geheim bleiben
Zugriffskontrolle Kontrolliert, dass jede Identität nur auf die Informationen und Dienste zugreift, zu welchen sie auch zugriffsberechtigt ist
Nicht Ablehnung Schütz davor, dass andere Einheiten nach einer Kommunikation behaupten können, nie daran teilgenommen zu haben

Wichtige Eigenschaften von Verschlüsselungsalgorithmen

Fehlerausbreitung: Charakterisiert die Effekte von Bitfehlern während der Übertragung von Ciphertext zum rekonstruierten Klartext Synchronisation: Charakterisiert die Effekte von verlorenen Ciphertexten auf den rekonstruierten Klartext

Sicherheitsziele von IPSec

Datenherkunftsauthentisierung/Datenintegrität maskierte Quell- oder Zieladresse zu versenden, Pakete während der Übertragung zu verändern, gespeichertes Paket zu späterem Zeitpunkt zu versenden soll unmöglich sein (dass der Empfänger dies nicht merkt)
Vertrauenswürdigkeit Es soll nicht möglich sein, den Inhalt der IP Datagramme auszuspähen; Es soll weiterhin eine begrenzte Traffic Flow Confidentiality geben
Sicherheitsrichtlinie Sender, Empfänger und zwischenliegende Knoten sollen erkennen können, ob ein Paket ihrer Sicherheitsrichtlinie entspricht und dieses gegebenenfalls verwerfen

Pakete

DHCP

DHCP Discover an Broadcast (255.255.255.255), Server sendet DHCP Offer zurück mit Payload, DHCP Request (gleich wie Discover)

DHCP Discover/Offer/Request/ACK
UDP/TCP SrcPort & DstPort
IP SrcIP & DstIP
MAC SrcAddr & DestAddr
Payload (optional)

ARP

ARP-Request/Response:

ARP ARP-Request/Response
MAC SrcAddr XYZ
DestAddr XYZ

Payload XXXXX

DNS

(A-Records bilden URL auf IP ab)

DNS
- DNS Query 'A random.org'
- DNS Response 'A random.org 123.45.67.890'

UDP/TCP SrcPort & DstPort
IP SrcIP & DstIP
MAC SrcAddr & DestAddr

Ports

UDP DHCP	67/68
FTP	21
SSH	22
Telnet	23
SMTP	25
DNS	53
IMAP	143
IMAP TLS/SSL	993
Non-privileg	>1023

TCP/IP

nicht existentes Modell, sehr nützliches Protokoll

Internetlayer Packetswitching, Adressierung, Routing und Forwarding. Insbesondere für hierarchische Netze

Transportlayer
- zuverlässiger Bytestrom: TCP (Transport Control Protokoll)
- unzuverlässiges Datagramm: UDP (User Datagramm Protokoll)

UDP

- minimalistisch
- Best Effort Dienst: Segmente können verloren gehen, nicht reihenfolgegetreu
- Verbindungslos: Kein Handshaking und unabhängige Behandlung der Pakete
- oftmals für das Streamen von Multimediainhalten
- Überprüfung durch Checksummen

TCP

- Punkt-zu-Punkt: Ein Sender, ein Empfänger
- Zuverlässiger, reihenfolgegetreuer Bytestrom
- Pipelined: Staukontrolle und Flusskontrolle
- Sende und Empfangspuffer
- Vollduplex Daten: Bidirektionaler Datenfluss
- Zuverlässsiger Datenverkehr benötigt eine Behandlung von Timeouts (RTT)

Schicht 2: HDLC (High Level Data Link Control)

3 verschiedene Stationstypen/Sendemodi:

- Primary Station (sendet Kommandos)
- Secondary Station (sendet Antworten)
- Combined Station (sendet beides)

Bei HDLC wird zur Unterscheidung des Endflags von den Übertragungsdaten Bitstopfen eingesetzt. Damit bezeichnet man den Vorgang, dass bei fünf aufeinanderfolgenden Einsen, stets eine Null eingefügt wird. Somit kann der Trailer, der 6 aufeinanderfolgende Einsen enthält, eindeutig unterschieden werden.

ISO/OSI - sehr nützliches Modell, keine existierenden Protokolle

Ein Protokoll arbeitet genau auf einer Schicht und beschreibt Syntax und Semantik der auszutauschenden Anwendungsinformation.

Desweiteren enthalten Protokolle Festlegungen bezüglich Antwortzeiten, Ablauffolgen und Dateneinheiten.
Jedes Layer nimmt Daten vom darüberliegenden Layer, fügt eine Headereinheit hinzu und erstellt eine neue Dateneinheit und schickt diese an das Layer darunter

Abstrakte Sicht Beschreibt den Aufbau eines Schichtenmodell im Allgemeinen
Funktionelle Sicht Beschreibt die Funktionen der sieben Schichten
Protokoll Sicht Beschreibt die Protokolle der einzelnen Schichten
Dienst Sicht Beschreibt die Dienste einer Schicht gegenüber einer höherliegenden Schicht

Schichtkommunikation

Eine Schicht (N) bietet der darüberliegenden Schicht $(N+1)$ an Dienstzugangspunkten (SAP - Service Access Point) Dienste an. Auf diese Dienste wird mit Hilfe von Dienstelementen (Service Primitives) zugegriffen. Man unterscheidet die vier folgenden Dienstelementetypen:

- Request (Anforderung)
- Indication (Anzeige)
- Response (Antwort)
- Confirm (Bestätigung)

Möchte eine Instanz einer Schicht $(N+1)$ in einem System A eine Verbindung zu einem System B aufbauen, muss sie bei der Schicht (N) im eigenen System den Dienst anfordern (Request). Für Schicht $(N+1)$ transparent baut Schicht (N) aus System A eine Verbindung zu Schicht (N) aus System B auf. Dabei benutzt sie gegebenenfalls Dienste der darunterliegenden Schichten. Schicht (N) in System B signalisiert nun (Indication) Schicht $(N+1)$ in System B das ein Dienst angefragt wurde. Schicht $(N+1)$ in System B antwortet Schicht (N) in System B (Response), wenn der Dienst akzeptiert wurde. Wiederum transparent für die darüberliegenden Schichten gibt Schicht (N) aus System B an Schicht (N) zurück das der Dienst akzeptiert wurde. Schicht (N) aus System A kann nun Schicht $(N+1)$ aus System A den Dienst bestätigen (Confirm).

Begriffe

ARP Adress Resolution Protocol Broadcast auf das LAN, mit der Frage, welcher Node IP X.X.X.X hat → Antwort des Nodes mit der MAC-Adresse → Zustellung möglich

Assymmetrische Kryptographie verwendet zwei Schlüssel für Ver- und Entschlüsselung

Authorative DNS Server DNS Server einer Organisation, stellen den authorativen Hostnamen für das IP Mapping der Organisationsserver

Baudrate bezeichnet die Schrittrate $\frac{1}{\delta t}$ mit der sich das Signal ändern kann. Pro Schritt kann eine Informationsmenge $ld(n)$ übertragen werden, wobei n die Anzahl der Quantisierungsstufen des Signals ist.

Bastion Host Ein Computer, welcher besonders gesichert werden muss, da er anfälliger für Angriffe ist, als andere Computer im Subnetz

Bedrohnung Eine Bedrohung in einem Kommunikationsnetzwerk ist jedes mögliche Ereignis oder eine Sequenz von Aktionen, welche zu einer Verletzung einer oder mehrerer Sicherheitsziele führen

Bitrate bezeichnet die übertragene Information in bit pro Schritt $[bit/s]$

BGP Border Gateway Protocol. Routerpaare tauschen Routinginformationen über semipermanente TCP Verbindungen aus

Bridge Jedes mit einer Bridge verbundene Netzwerk ist eine eigene Kollisionsdomäne und auch verschiedene LAN-Typen können miteinander verbunden werden

Broadcast ein Sender, alle Teilnehmer eines Netzes; Adressierung an alle

Broadcastkanal Völllig dezentralisiert und so einfach wie möglich mit Rate b/s

Broadcast Medium Nur ein Sender zu jeder Zeit; Zugriffskontrolle (MUX o. Absprache)

Burst Traffic

Bustoplogie Alle Geräte sind an einem Kabel angebunden und sind in einer Kollisionsdomäne

Cipher Methode eine Nachricht so zu transformieren, dass die Bedeutung nicht mehr erkannt werden kann

Client Kommunizieren zeitweise mit Server; Können dynamische IP-Adressen haben; Kommunizieren nie direkt miteinander

CSMA Carrier Sense Multiple Access

CSMA/CD + Collision Detection

CSMA/CA + Collision Avoidance

Circuit Switching einfach; einmal aufgesetzt verbleiben die Ressourcen beim Nutzer; Circuit muss hergestellt werden, bevor kommuniziert werden kann

Delay d = distance / speed v

DHCP Dynamic Host Configuration Protocol. beziehe die Adresse dynamisch von einem Server

Dual Homed Host Ein Computer mit ¿ 2 NetzwerkinterfacesProxies leiten genehmigte Clientanfragen an die Server, und die Antworten auch wieder an den Client weiter

Effizienz Definiert als die Rate der Zeit, in welcher der Sender neue Informationen sendet (für den fehlerfreien Kanal)

Fehlerkontrolle vorwärts Sender sendet redundante Infos so, dass der Empfänger selbst ausbessern kann

Fehlerkontrolle rückwärts Sender sendet redundante Infos so, dass der Empfänger fehlerhafte Pakete wahrscheinlich erkennt und Pakete in dem Fall nochmal verschickt werden können

Firewall Eine oder eine Menge an Komponenten, welche den Zugriff zwischen einem geschützten Netzwerk und dem Internet oder zwischen einer Menge an Netzwerken beschränkt

FTP File-Transfer-Protokoll: Dateitransferprotokoll, Übertrage Daten von und zum Server

Full Duplex Übertragung gleichzeitig in beide Richtung (Frequency/Time Division Duplex)

Gateway Router Spezielle Router innerhalb des AS, führen das Intra-AS Routingprotokoll mit allen anderen Routern im AS aus. Zusätzlich verantwortlich für das Routing an exteren Ziele → Inter-AS Routingprotokolle mit anderen Gatewayroutern

Half Duplex Übertragung abwechselnd in beide Richtungen (Time Division Duplex)

Hammingdistanz Anzahl an Stellen an denen sich zwei Frames x und y in binärer Darstellung unterscheiden lösbar mittels (x XOR y)

Hot Potato Routing Wenn ein Paket ankommt, so leite es auf schnellste Art und Weise an den Ausgang mit der kleinsten Ausgangswarteschlange, ganz egal wohin dieser Ausgang dann führt

HTTP Hyper Text Transfer Protocol; Das Anwendungsnachrichtenprotokoll des Webs

Hub Eingehende Bits werden an alle Ausgänge mit selber Rate und ohne Puffern verteilt; Kein CSMA-CD am Hub; Alle verbundenen Kabel formen eine Kollisionsdomäne

IMAP Internet Message Access Control

IPSec Authentication Header (AH) Im Tunnelmodus stellt der Payload nochmals ein ganzes IP Paket dar; Wichtig: AH funktioniert nur in NAT freien Umgebungen

IPSec Encapsulating Security Protocol (ESP) Dem ESP Header folgt direkt ein IP Header oder ein AH-Header; Das next-header Feld vom vorhergehenden Header indiziert 50 für ESP

Kryptologie Wissenschaft, die sich mit Kommunikation in sicherer und geheimer Art befasst

Kryptographie (graphein = schreiben): Die Lehre der Prinzipien und Techniken, durch welche Informationen in Ciphertext verpackt und später durch legitimierte Nutzer, wieder durch einen geheimen Schlüssel entschlüsselt werden können

Kryptoanalyse (analyein = etwas lösen): Die Wissenschaft und Kunst Informationen von Ciphern wiederherzustellen und dies ohne das Wissen über den Schlüssel zu schaffen

Link State Routing Berechnung des kleinsten Kostenpfades von einem Knoten S zu allen andern Knoten V erzielt durch den Link-State-Broadcast

Lokal DNS Server Jeder ISP hat einen eigenen; Wenn ein Host eine DNS Anfrage stellt, so wird die Frage zuerst zum lokalen DNS Server gesendet (fungiert also als ein Proxy)

Medium Access Control (MAC) Verteilter Algorithmus, der bestimmt, wie Knoten auf ein geteiltes Medium zugreifen

Mail Useragent Erlaubt das Schreiben, Lesen und Bearbeiten von Nachrichten; Ein- und ausgehende Nachrichten werden auf einem Server gespeichert

Mailserver Die Mailbox beinhaltet eingehende Nachrichten, die Nachrichtenschlange die ausgehenden Nachrichten

Multicast Ein Sender, eine Gruppe von Empfänger; Adressierung an eine Gruppe bekannter Adressen

MIME Multimedia Mail Extensions: Zusätzliche Zeilen im Nachrichtenheader deklarieren den MIME Inhaltstyp

Network Address Translation (NAT) eine Prozedur, durch welche ein Router die Daten in Paketen ändert um die Netzwerkadressen zu modifizieren; Dies erlaubt es die interne Netzwerkstruktur zu verschleiern

Nichtpersistentes HTTP höchstens ein Objekt wird über die TCP Verbindung verschickt

OSPF Open Shortes Paths First. annocieren nun keine Wege sondern Linkzustände mit je einem Eintrag pro Nachbarknoten

Packet Switching Aufteilen von Daten in kleinere Pakete die nach und nach gesendet werden; Problem: Informationen zu Sender/Empfänger und Start/Endzeitpunkt eines Pakets müssen mit übermittelt werden; Wird deshalb 'Store and Forward' Netzwerk genannt

Paketfiltern/Screening Die Aktion, welche ein Gerät ausführt, um selektiv den Fluss an Daten in und aus einem Netzwerk zu kontrollieren. Paketfiltern ist eine wichtige Technik um Zugriffskontrolle auf dem Subnetzwerklevel für paketorientierte Netzwerke zu implementieren

Peer to Peer Ohne ständig eingeschalteten Server. Beliebige Endsysteme kommunizieren direkt miteinander, sind dabei zeitweise verbunden und haben wechselnde IP Adressen

P2P Filesharing Ein Peer ist sowohl ein Webclient als auch ein transienter Webserver; Alle Peers sind Server → Hoch Skalierbar; Dateiübertragung ist dezentralisiert, die Lokalisierung findet allerdings zentral statt.

Perimeternetzwerk Ein Subnetz, welches zwischen einem externen und einem internen Netzwerk hinzugefügt wird, um eine weitere Sicherheitseben bereitzustellen; Ein Synonym hierfür ist DMZ (De Militarized Zone)

Persistentes HTTP Mehrere Objekte können über eine TCP Verbindung zwischen Client und Server ausgetauscht werden

Point-to-Point Adressierung an eine bekannte Adresse

Poisoned Reverse Wenn Z durch Y routet um zu X zu gelangen: Z sagt Y, dass seine eigene Distanz zu X unendlich ist (somit routet Y nicht über X nach Z)

Polling Masterknoten läd Slaveknoten zum Übertragen in Reihenfolge ein

POST Methode Webseiten beinhalten oft Formulareingaben, die Eingabe wird dann im Entity Body an den Server geschickt

Protokoll Protokolle sind Regelsätze, welche beschreiben wie zwei oder mehr entfernte Teile (peers oder protocol entities) eines Layers kooperieren, um den Dienst des gegebenen Layers zu implementieren. Ein Protokoll ist die Implementierung eines Services

Proxy ein Programm, welches sich im Auftrag interner Clients mit externen Servern beschäftigt

QPSK Quadrature Phase Shift Keying; Phasenverschiebung für Multiplexing

Repeater Physical Layer Gerät, verbindet zwei Kabel und verstärkt die ankommenden Signale und leitet dieses weiter; Versteht den Inhalt der Pakete nicht und interessiert sich nicht dafür

Ressource Records (RR) in DNS Datenbank; Format: (name, value, type, ttl)

RIP Routing Information Protocol. Distanzvektoralgorithmus mit Hops als Metrik. Falls nach 180s kein Advertisement empfangen wurde, so deklariere den Nachbarn als tot

Routing Berechnen der Route, die die Pakete von Quelle bis zum Ziel gegangen sind

RTT Round Trip Time: Benötigte Zeit um ein kleines Paket so zu senden, dass es vom Client zum Server und zurück geschickt wird

Rückwärtslernen (Routing) Paketheader enthalten wichtige Infos, wie Quelle, Ziel, Hopzähler → Netzwerkknoten lernen etwas über die Netzwerktopologie während sie Pakete behandeln

Server ständig eingeschaltet und mit permanenter IP-Adresse; Serverfarmen zur Skalierung

Simplex Übertragung in eine Richtung

Signale sind die physische Repräsentation von Daten in der Form einer charakteristischen Variation in Zeit oder Ausbreitung...

Signieren von Daten Berechnet einen Checkwert oder eine

digitale Signatur zu einem gegebenen Plaintext oder Ciphertext, sodass dieser durch alle oder einige Instanzen mit Zugriff verifiziert werden kann

SMTP Mailübertragungsprotokoll: Verwendet TCP um Nachrichten zuverlässig vom Client zum Server zu übertragen, verwendet Port 25; Direkte Übertragung vom Sender zum Empfänger

Socket Ein lokal auf dem Host laufendes, von einer Anwendung erstelltes, OS-kontrolliertes Interface, durch welches ein Anwendungsprozess sowohl Nachrichten vom und zu anderen Anwendungsprozessen Senden, als auch Empfangen kann

Statisches Multiplexing einzelne Ressource statisch gemultiplext durch feste Sendezeiten und mehrere Frequenzbänder

Sterntopologie einfachere automatische Verwaltung und Wartung bei fehlerhaften Adaptern

Strict Layering Jedes Layer verwendet nur den Service des darunter liegenden Layers

Spannbaum Gegeben sei ein Graph G=(V,E), ein Spannbaum T = (V,E-T) ist ein Subgrap von V, wobei E-T ein Teil von E ist, welcher ein Spannbaum, der verbunden und azyklisch ist

Switch nicht nur eine einfache elektrische Verbindung für sternförmige Topologie; Switches enthalten Puffer, welche direkt ankommende Pakete zwischenspeichern, bevor sie diese weiterleiten

Symmetrische Kryptographie verwendet einen Schlüssel für Ver- und Entschlüsselung oder Signieren und Überprüfen

TCP Zuverlässige, in-Order Zustellung, Stau- & Flusskontrolle, Verbindungsaufbau

TLP Server Top Level Domain Server: Verantwortlich für .com, .org, .net, .edu und die Landesdomains

Tokenweitergabe Kontrolltoken wird von einem zum anderen Knoten übertragen

UDP Unzuverlässige, ungeordente Zustellung, Einfache Erweiterung des best Effort IP Ansatzes

Unicast Ein Sender, ein Empfänger

URL Methode Verwendet die GET Methode; Die Eingaben werden im URL Feld der Requestline hochgeladen

Verschlüsseln von Daten Transformiert Plaintext in Ciphertext um die Inhalte zu verschleiern

Weiterleiten Bewege Pakete vom Routereingang auf den entsprechenden Ausgang

Formeln

Bitzeit $t_{Bit} = \frac{1}{Bitrate}$

Bitlänge $l_{Bit} = v_s * t_{Bit}$

Ausbreitungsverzögerung $d_{prop} = \frac{dist}{v_s}$

Übertragungszeit $d_{trans} = \frac{L}{R} = [\frac{bit}{s}]$

Ende-zu-Ende-Verzögerung $d_{e2e} = d_{prop} + d_{trans}$

Leitungsverm. Übertragung $t_L = \frac{L_{Nachricht}}{R}$

Nachrichtenver. Übertragung $t_N = (k+1)\frac{L_{Nachricht}}{R}$

Paketver. Übertragung $t_P = (k + \frac{Laenge_{Nachricht}}{Laenge_{Pakete}}) * \frac{L_{Packet}}{R} = (1 + \frac{k}{n}) * \frac{L_{Nachricht}}{R}$

Kanalkap. Nyquist $R_{max} = 2 * H * log_2 n$

Kanalkap. Shannon $R_{max} = H * log_2(1 + \frac{P_signalleistung}{P_rauschleistung})$ mit $r = 10 * log_{10} * \frac{P_s}{P_n}$

Fehlerfrei Send and Wait $S = \frac{1}{(1+2a)}$ wobei $a = \frac{T_{prop}}{T_{trans}}$

Fehlerhaft Send and Wait $S = \frac{1-P}{1+2a}$

Fehlerfreies Sliding Window $S = 1, fallsW >= 2a+1, W/(2a+1) sonst$

Selective Reject $S = 1-P, fallsW >= 2a+1, (W(1-P))/(2a+1) sonst$

Go-Back-N $S = \frac{1-P}{1+2aP}, fallsW >= 2a+1, \frac{W(1-P)}{(2a+1)(1-P+WP)} sonst$

Effizienz $\frac{T_{packet}}{T_{packet}+d+T_{ack}+d}$

efficiency $\frac{1}{(1+5*\frac{t_{prop}}{t_{trans}})}$

Round Trip Time $EstimatedRTT = (1-a) * EstimatedRTT + a * SampleRTT$

TCP Durchsatz $0,75 * \frac{W}{RTT}$

Konkrete Netzwerke und Protokolle

Hardwareschicht

Realisierung	10Base2/5 Ethernet	bis 1GBit Ethernet	Token-Bus	Token-Ring	DQDB	FDDI
Protokoll IEEE	CSMA/CD 802.3	CSMA/CD 802.3	Token-Bus 802.4	Token-Ring 802.5	Distributed Queue Dual Bus 802.6	Fiber Distributed Data Interface -
physische Topologie logische Topologie	Bus Bus	Bus Stern	Bus/Baum Ring	Ring/Stern Ring	Bus/Ring Bus	Ring+ Ersatz Ring
Leitungskodierung Bitrate Baudrate	Biphase-L 10 Mbit/s 20 MBaud	unterschiedlich 10-1000 Mbit/s 20-2000 MBaud	1,5/10 Mbit/s	Differential 1-1000 Mbit/s	44,736 Mbit/s	4B/5B, NRZI x $1,25 * x$
Kanalzugriff Anwendungsgebiet	CSMA/CD LAN	CSMA/CD LAN	Token Zeitkritische Systeme	Token Zeitkritische Systeme	DQDB-Zellen MAN	Token MAN

ISO/OSI Layer

	Layer	Dateneinheit	Aufgaben	Typische Spezifizierungsaufgaben in dieser Schicht
PH	Physisch	bit	• Senden und Empfangen von Folgen von Bits	• Definition des Mediums • Festlegung der Spannungsbereiche von 0 und 1 • Festlegung der Übertragungsdauer eines Bits • Festlegung der Anzahl der Leitungen und ihrer Nutzung • Zeitliche Synchronisation (Non-Return to Zero Level oder Manchstercodierung) • Breitband- vs Basisbandübertragung (Amplituden-/Phasen-/Frequenzmodulation)
L	Link	Datenrahmen	• Senden und Empfangen von Datenrahmen • dazu Generierung und Erkennung von Rahmenbegrenzern • Bearbeitung von Bestätigungsrahmen • Flussregelung • Fehlererkennung	• Festlegung der Rahmenbegrenzer • Festlegung des Fehlerbehandlungsverfahrens • Festlegung der Möglichkeiten der Flusssteuerung • Im Fall von Halb-duplex oder multipoint links muss der Zugriff auf das Medium kontrolliert werden und Peersysteme müssen addressiert werden. • Framing durch Charakterzählen, Flagbitmuster/Bitstuffing oder Codeverletzung • Fehlererkennung & -kontrolle (vorwärts/rückwärts) mit Redundanz (Parität), Hemmingdistanz, Cyclic Redundancy Check (CRC) • Send and Wait (Sliding Window) , Go-Back-N, Selective Reject
N	Network	Datagramm, Paket	• Routing des Datenverkehrs • Absicherung von Verbindungsqualitäten • optional: Abrechnungsfunktion	• Der N-Service ist uniform, unabhängig von der Variation an Subnetwork Technologien, Topologien, QoS und der Organisation • Netzwerk Addresse = Endsystem Addresse
T	Transport	TPDU (Transport Protocol Data Unit)	• Aufbau virtueller Point-to-Point-Verbindungen • Senden im Broadcast und Multicast Modus • optional: Aufteilen der Transportverbindung auf mehrere Netzverbindungen • optional: Multiplexen mehrerer Transportverbindungen auf einer Netzverbindung	• Sendeseite: Segmentiert Anwendungsnachrichten und leitet diese Segmente an die Netzwerkschicht • Empfangsseite: Reassembliert Segmente in Nachrichten und leitet diese an die Anwendungsschicht weiter • Fehlerkontrolle: Durch Sequenznummern, ACKs und Neuübertragungen • Flusskontrolle: Durch Inspizieren von ACKs und Permits • Staukontrolle: Durch das Verlangsamen des Senders, wenn Pakete oder ACKs verloren gehen
S	Session	SPDU (Session Protocol Data Unit)	• Aufnahme und Entgegennahme einer Sitzung • z.B. Authentifizierung an einem anderen Rechner • Bereitstellen von Synchronisierungspunkten zur Wiederaufnahme von abgebrochenen Verbindungen	• Quarantine Data delivery - Eine ganze Gruppe von übertragenen S-SDUs wird zugestellt auf explizite Anfrage des Senders • Interaktionsverwaltung erlaubt ausdrücklich festzulegen, welcher S-User das Recht bekommt zu übertragen
P	Presentation	PPDU (Presentation Protocol Data Unit)	Konvertierung der Daten in das Netzdatenformat	• Ausschließlich die Syntax wird modifiziert um die Semantik beizubehalten
A	Application	APDU (Application Protocol Data Unit)	Bereitstellung anwendungsunterstützender Elemente	• Unterstützt den direkten Endnutzer durch die Bereitstellung einer Vielzahl an application services • Genereller Typ (z.B. Entfernte prozedurale Anrufe, Transaktionsdurchführung,...) • Spezifischer Typ (z.B. Virtuelles Terminal, Dateiübertragungszugriff und Verwaltung, Arbeitswechsel,...)