1. tétel

Számítógép hálózatok kialakulásának okai, története. Az Internet kialakulása. Hálózatok osztályozása kiterjedésük, topológiájuk, az összeköttetés típusa, a kapcsolási módok és az információátvitel iránya alapján.

A gyors adatátvitel, ill. a nagyobb teljesítmény elérése érdekében a számítógépeket egy közös kommunkiációs rendszerben kapcsolják össze. Az összekapcsolt gépeket munkaállomásoknak (usereknek) nevezik. Ezeket a hálózatba kötött számítógépeket, egy speciális ún. ***hálózati operációs rendszer*** működteti. A számítógép-hálózat számítógépei a rendszerben egymással adatokat, információkat cserélhetnek, ill. "*erőforrásaikat megosztva*" használhatják. Ilyen *erőforrások* lehetnek a winchesterek, nyomtatók, programok, de lehetnek könytárak és azok állományai is. A *megosztás* pedig annyit tesz, hogy az adott munkaállomás tulajdonosa, hozzáférési jogosultságot ad a saját gépének winchesterén lévő, általa kiválasztott és megjelölt könyvtárába való betekintésre. Az információcserét ún. hálózati vezérlőkártyák és adatkábel rendszer biztosítja. A számítógép-hálózatok mivel több számítógépből állnak, sok esetben igen nehéz és költséges azonos típusú és konfigurációjú számítógépből felépíteni. Több géptípus egyenes következménye a kompatibilitási probléma Ezért a hálózat tervezésekor ezt mindenkor figyelembe kell venni, a megfelelő szabványok és protokollok használatának alkalmazásával.

Az Internet kialakulása és fejlõdése napjainkig

A hidegháborús versengés egy fontos problémát hozott elõtérbe Amerikában a hatvanas évek elején. Az amerikai kormányszervek hogyan tudják megtartani a kommunikációt egy esetleges nukleáris háború alatt, illetve után? Az országnak egy olyan kommunikációs rendszerre volt szüksége amely összeköti egymással a hivatalokat, bázisokat. Az egyik legfontosabb szempont az volt, hogy a hálózat minden körülmények között megõrizze mûködõképességét. Az egyes egységek megsemmisülése ne vonja magával az egész rendszer összeomlását. A megoldást Paul Baran vezérkari tiszt szolgáltatta. Ezt a tervet elõször 1964-ben hozták nyilvánosságra. Az alapelvek meglehetõsen egyszerûek voltak:

- A hálózatnak nem lehet semmiféle központja, úgy kell mûködtetni mintha darabokból állna. Minden egyes csomópont egymással egyenértékû.
- A hálózatot mindenkor megbízhatatlannak kell feltételezni.
- Az üzenetek csomagokra oszlanak. Minden egyes csomagnak saját címe van.

Az hogy a csomag milyen úton jut el a címre teljesen lényegtelen. Az adó feldarabolja az üzenetet és elküldi a vevõ irányába. A vevõ képes érzékelni, ha valamelyik csomag nem érkezik meg és azt újra lekéri. Az hogy a csomagok milyen útvonalon érkeznek meg teljesen lényegtelen.

Csoportosításuk kiterjedés szerint:

Három nagy csoportját különböztetjük meg:

Helyi hálózatok (LAN - Local Area Network***)***: olyan rendszerek, amelyekben a számítógépek fizikailag viszonylag egymáshoz közel helyezkednek el, például egy épületen belül.

Nagyterületi hálózatok (WAN - Wide Area Network***)*** :ezek olyan rendszerek, melyeknek egyes szegmensei (elemei) földrajzilag is távol lehetnek egymástól. Ebben az esetben a kapcsolattartás más speciális módszerekkel valósítható meg.

Globális hálózatok:ezek a rendszerek olyan világméretű hálózati rendszerek, melyek nagyszámú elemet tartalmaznak, eléggé heterogén felépítésűek, nagyon sok számítógépet, ill. részhálózatot foglalnak magukba. Az Internet az élő példa.

A végrehajtandó feladatok szerint is megkülönböztethetjük a hálózatokat. A legegyszerűbbek az ***egyenrangú***, ún *peer to peer* hálózatok. A legelterjedtebb azonban a ***szerver-kliens*** felépítésű hálózat, melyben a *szerver* kitüntetett szereppel bír. A szerver látja el a hálózati rendszerek vezérlését és felügyeletét, míg a munkaállomások (kliensek, userek) a szerverhez kapcsolódva, annak irányítása alatt képesek használni a hálózatot.

Felépítésük (topológiájuk)

A hálózat felépítését, topológiáját a kábelek elrendeződése, a csomópontok fizikai elhelyezkedése határozza meg. Ez a "hálózat alakja". (az Ethernet lineáris, vagy sín hálózati, ill. csillag topológiát alkalmaz)

Sín: a hálózatnak van egy gerince(BackBone - közös adatátviteli vonal), amihez az összes csomópont csatlakozik. A gerinc mindkét vége ellenállással van lezárva, a rendszer elemei sorba vannak fűzve egy kábelre. Minden csomópontnak egyedi címe van. Olcsó, kevés kábel kell hozzá. Hiba esetén az egész hálózat működésképtelen lesz.

Csillag: a csomópontok egy közös elosztóba (hub) vannak bekötve. A csillag topológiánál ilyen elosztók gyűjtik össze egy-egy gépcsoport jeleit és továbbítják a központ felé. A csillag topológia előnye az, hogy egy új elosztó beépítésével újabb és újabb gépcsoportokat lehet a rendszerhez kapcsolni. Nem üzenetszórásos (ponttól-pontig). Szakadás esetén megbízhatóbb, sok kábel kell hozzá ezért drága.

Gyűrűs : a csomópontokat közvetlenül egymáshoz csatlakoztatják, soros elrendezésben, így azok egy zárt hurkot alkotnak. Az üzenetek fogadása egy alkalmas csatoló eszköz segítségével történik. Előre történő huzalozása nehézkes, új csomópont hozzáadása, vagy elvétele megbonthatja a hálózatot. A biztonság kedvéért 2 kábellel is összeköthetik a gépeket. Az adatáramlásnak meghatározott iránya van. Amíg az adatot nem mentik le, addig a gyűrűben kering, tárolódik. Nagy a kockázat, az adatok sérülhetnek, elveszhetnek. Ezt elkerülendő a címzettnek mielőbb le kell menteni és nyugtázni, hogy ne keringjen a végtelenségig.

Hierarchikus (fa): A busz topológia fa topológiává egészíthető ki, amelyban a többszörös buszágak különböző pontokon kapcsolódnak össze, így alkotva egy fastruktúrát. Meghibásodás esetén csak a csomópont és a hozzátartozó gyökerek esnek ki.

Vegyes: az előző formák vegyes alkalmazása. ...

Hálózati összeköttetések típusai

Összeköttetés alapú (virtuális áramkör)

csomagok állandó útvonalon a forrás és cél között, virtuális áramkör az összeköttetés felépítésétől a bontásig a forgalomszabályozás az összeköttetés része pl.: telefon

Összeköttetés mentes

a datagrammok útválasztása egymástól független, a csomagok a forrás és a cél teljes címét tartalmazzák egy meghibásodott csomópont nem okoz komoly gondot pl.: e-mail

Kommunikáció iránya szerint

Simplex (csak egyirányú) Az egyik állomás csak az adó a másik csak a vevõ.

Fél duplex (váltakozó irányú) Mindkét irányban megengedett az adatátvitel, de egy idõben csak az egyik irányban élhet.

Duplex (kétirányú) Mindkét állomás egyszerre lehet adó és vevõ is.

Kapcsolási technika alapján

Vonalkapcsolt A kommunikáló állomások között állandó kapcsolat épül ki az adás idejére. Jó példája a telefon.

Üzenetkapcsolt A két állomás között az átviteli hálózat tárolva továbbító - store-and-forward - számítógépekbõl áll, ezek továbbítják az üzeneteket egy címinformáció alapján. Az üzenet hossza nem korlátozott. Hasonlít a postai csomagküldéshez.

Csomagkapcsolt Hasonlít az üzenetkapcsolthoz, csak a csomag mérete maximált, ezért az üzeneteket **csomag**okra (packet) kell darabolni.

2. tétel

A sávszélesség és az átbocsátóképesség fogalma. Az adatátviteli sebesség elvi korlátai. Vezetékes átviteli közegek (coax és CAT kábelek, optikai szálak) felépítése, működése, átviteltechnikai jellemzői.

Sávszélesség: analóg rendszerek esetén használt fogalom: egy adott analóg jel maximális és minimális frekvenciájának a különbségét értjük alatta.

Adatátviteli sebesség: az időegység alatt átvitt bitek száma. Ezt célszerű bit/s-ban (bps) mérni. Az átvitelt jellemezhetjük a felhasznált jel értékében 1 másodperc alatt bekövetkezett változások számával is, amit jelzési sebességnek, vagy közismert néven **baud**-nak nevezünk.

Az **átbocsátóképesség** fogalma a tényleges, mért sávszélességre utal, amely a nap egy adott időpontjára, meghatározott internetes útvonalakra, meghatározott adatok átvitelére vonatkozik. Különböző okok miatt az átbocsátóképesség sajnos gyakran jelentősen elmarad az adott átviteli közegre jellemző maximális digitális sávszélességtől. Többek között az alábbi tényezők határozzák meg az átbocsátóképességet:

- A hálózat-összekapcsoló készülékek
- Az átvitt adatok típusa
- A hálózati topológia
- A hálózaton levő felhasználók száma
- A felhasználó számítógépe
- A kiszolgáló számítógép
- Az áramellátás

Az aktuális adatsebesség

- Kisebb, mint az átviteli közeg sávszélessége
- Értékét meghatározza:

- Sávszélesség
- A hálózat terheltsége (más felhasználók)
- Az átvitel zavaró tényezői (zajok)
- Végpontok

Kiszolgáló

Kliens

- A hálózatban alkalmazott eszközök
- A hálózat átviteli közegei
- Topológia, protokollok, útválasztás…

Az adatátviteli sebesség elvi korlátai:

Nyquist(1924) bizonyította

–Ha tetszőleges jelet H sávszélességű alul áteresztő szűrűn átengedünk akkor a szűrt jelből másodpercenként 2H-szor mintát véve az eredeti jel teljesen visszaállítható. Ebből:

–Max_adatátviteli_sebesség= $2\ H \log_2 V$

Ahol H: a csatorna sávszélessége, V: a jel diszkrét értékeinek száma (jelzések száma).

(Azaz V érték $\log_2 V$ bitet hordozhat.)

Shannon-tétel

C. Shannon(1948) határozta meg a véletlen (termikus) zajjal terhelt csatornákra az elméleti maximális adatátviteli sebességet (információelméleti megfontolások alapján)

•Max elérhető adatátviteli sebesség = H log2(1+S/N)

ahol

–H: a csatorna sávszélessége;

–S/N: a jel-zaj viszony (signal-tonoiseratio)

- •S: jelteljesítmény;
- •N: zajteljesítmény

Lehetséges átviteli közegek:

- **Vezetékes átvitel:** - Rézvezeték: Coax, UTP, STP
  - Optikai szál: Mono- és multi módus
- **Vezeték nélküli átvitel:** - Látható vagy nem látható fény: IrDA, lézer
  - Rádióhullámok: WiFi
  - Optikai vagy mágneses hordozó

Vezetékes átviteli közegek

Csavart érpár (UTP, STP)

A legrégebbi és még ma is elterjedt átviteli közeg a csavart érpár vagy más néven **sodrott érpár** (**U**nshielded **T**wisted **P**air = **UTP**). Egymásra spirálisan felcsavart rézvezetékek. Minél sûrûbb a csavarás, annál nagyobb az adatátviteli sebesség. Az STP kábelek esetében az árnyékolást egy, a koaxiális kábeléhez hasonló rézháló biztosítja, míg az FTP esetében ez egy

vékony fémfólia köpenyt jelent a sodrott érpárak körül. Az STP kábelek sokkal vastagabbak, nehezebben telepíthetők, mint az FTP kábelek, árnyékolási paramétereik pedig rosszabbak, főként a nagyobb frekvenciák tartományában. A sávszélesség a huzalok vastagságától és az áthidalni kívánt távolságtól függ, de akár a Gbit/s-os nagyságrendû sebesség is elérhetõ.
A sodrott érpáras kábel nem lépheti túl a 100 méteres hosszúságot a hub és a számítógép között.
Egy irodában rengeteg berendezés kelthet zavart. Például a villanymotorok (gépek ventillátora, szellõztetõ berendezések stb.) különbözõ rádiófrekvenciás alkalmazások (pl.: mobiltelefon), valamint az elektromos hálózat. Szerencsés esetben ez csak a rendszer sebességére van kártékony hatással, rosszabb esetben hibás adattovábbítást, adatvesztést okoz.
Említést kell tenni a patch panelrõl és a patch kábelrõl. A patch panel egy olyan segédtábla, amely UTP-s hálózatoknál a felhasználói gépek felõl bejövõ kábelek rendezését szolgálja. A patch kábel viszont egy olyan viszonylag rövid, sodrott érpárú, UTP csatlakozóval ellátott kábel, amely a fali hálózati csatlakozó és a számítógép hálózati kábelének csatlakozója közti összeköttetést biztosítja.
A szabványügyi intézetek (EIA/TIA) több kategóriába sorolják a kábeleket (CAT 4,5,6 stb., átviteli MHz és Mbit/s alapján)

Koaxiális kábelek

Egy tömör rézhuzalból áll, amely körül szigetelõ van. A szigetelõt egy külsõ hengeres vezetõ veszi körbe, amelyet egy védõ mûanyagburkolat zár körül. Felépítésének köszönhetõen nagyon védett zajokkal szemben, és hosszú távú átvitelre is alkalmas. Könnyen meghosszabbítható, a különféle kábeltoldók, szétválasztók, csatolók és jelismétlõk segítségével. Két fajta koaxiális kábel létezik:

- Alapsávú: 50 ohm -os kábel, digitális átvitelt tesz lehetõvé
- Szélessávú: 75 ohm -os kábel, analóg átvitelt tesz lehetõvé

Üvegszálas kábel

Üvegszálas hálózat kiépítésére akkor kerül sor, ha különösen nagy elektromágneses hatások érik a vezetékeket vagy nagy távolságokat kell áthidalni. Itt a fényáteresztõ anyagból készült optikai szálon tovahaladó fényimpulzusok szállítják a jeleket. Az optikai kábel egy olyan vezeték, amelynek közepén üvegszál fut. Ezt az üvegszálat gondosan kiválasztott anyagú burkolat veszi körül. A különleges anyag tulajdonsága, hogy az ide-oda cikázó fény sohasem tudja elhagyni a kábelt. Ezért a fény a vezeték elején lép be és a végén lép ki belõle. De így is meg kell erõsíteni és újra kell rendezni a fényt. A legnagyobb áthidalható távolság manapság 80 kilométer. Az adó, ami lehet LED vagy lézer, elektronikus adatot küld át a kábelen melyet elõzõleg fotonná alakítottak.
Az optikai átviteli rendszer három komponensbõl áll: az **átviteli közeg**-bõl (hajszálvékony üveg vagy szilikát), amit egy szilárd **fénytörõ réteg** véd (szintén üveg vagy mûanyag), a **fényforrás**-ból (LED vagy lézerdióda) és a **fényérzékelõ**-bõl (fotodióda).

3. tétel

A vezeték nélküli adatátvitel lehetséges megoldásainak (látható és nem látható fénnyel történő átvitel, rádiófrekvenciás átvitel) bemutatása, előnyök, hátrányok ismertetése. Az IrDA és Bluetooth szabványok részletes bemutatása.

Vezeték nélküli adatátvitel

Vezeték nélküli átvitelt használó eszközök:- Kis hatótávolságú adóvevők, - Kis energiafogyasztás - Kis helyigény
Jól használható megoldás – Mobiltelefonokban, - Kéziszámítógépekben, - Notebookokban, - Fejhallgatókban, - Távirányítókban, - Nyomtatókban

Vezeték nélküli átviteli közeg

Hálózat kiépítésekor gyakran adódik olyan helyzet, amikor vezetékes összeköttetés kialakítása lehetetlen. Utcákat kellene feltörni, ott árkokat ásni és ha mindez mondjuk egy forgalmas, sűrűn beépített terület? Ilyenkor a vezeték nélküli átviteli megoldások közül kell választani, amelyek fény(infravörös, lézer) vagy rádióhullám alapúak lehetnek.

Lézer, infravörös átvitel

Külső helyszíneken rendkívül gyorsan telepíthető, nehezen lehallgatható. Az infravörös eszközök többsége napfényben nem használható. Az állomások telepítésénél gondosan kell eljárnunk, mert az adó kis mozgása, vagy a fény eltérítése (pl. a felmelegedő falról felszálló meleg levegő eltérítheti) az összeköttetés megszakadásához vezet.
A számítógépes rendszerekben az információátvitel ilyen módja fokozatosan terjed, IrDA néven már szabványos megoldása is létezik.
IrDA (Infrared Data Association)

- átvitel fény segítségével
- hatótávolság: kb. 5-10 méter
- kis teljesítményű verziók hatótávolsága kb. 20 cm
- adatátviteli sebesség: akár 4 Mbit/s (max.: 1 méter)
- rádiófrekvenciás zavarásra érzéketlen
- Kizárólag akadálymentes környezetben használható
- Csak pont-pont összeköttetésre használható

Rádiós összeköttetés

Nagy távolságok hidalhatók át közbenső állomások nélkül. Az átviteli sebesség relatíve kicsi. Az összeköttetést légköri zavarok, ionoszféra zavarok befolyásolják. Fő előnye az állomások mozgékonysága. Kis távolságok esetén a GHz tartományban működő rendszerekkel 10Mbit/sec sebesség könnyen elérhető. Nem kell kábelezni, gyorsan telepíthető. Az utóbbi időben rohamosan terjednek a néhány méter hatósugarú rendszerek. A cél a kábelek elhagyása, kényelmes használat szobán belül. (Telefon - számítógép – nyomtató, számítógép – egér, stb.) A mikrohullámú tartományban (2-40 GHz) irányított antennákkal megbízható, nagysebességű összeköttetések hozhatók létre. Az állomásoknak optikailag látni kell egymást. A telepítéskor tornyok, vagy magasan fekvő pontok szükségesek. Főként a magasabb frekvenciatartományokban az eső és hóesés jelentősen ronthatja az átvitelt, ezért helyettesítő útvonalakat terveznek be. A szokásos távolság az állomások között 30 – 100 km. Nagyobb távolságok esetén közbülső, reléállomások beiktatása szükséges. Rendkívül gazdaságos, versenyképes megoldás. Egyedüli hátrány, hogy most már hiány van kiosztható frekvenciasávokban. A frekvenciasávok kiosztása átgondolást igényel, és hatósági feladat.

Bluetooth

- kis hatótávolságú, a gyakorlatban is elterjedt ad-hoc hálózat
- cél: irodai, szobai eszközök közti összeköttetés vezeték nélkül (PC, nyomtató, telefon, szórakoztatóelektronika, stb)
- átvitel rádióhullámok segítségével
- nem igényel "rálátást"
- Maximális átviteli sebesség 1 Mbit/s
- Frekvenciasáv: 2,4 GHz
- 79 vivőfrekvencia (1 MHz-es csatornaosztás)
- 1600 (ál)véletlenszerű frekvenciaugrás másodpercenként
- Mester - szolga (Master - Slave) viszony
- Időosztásos duplexelés

Mester: minden páratlan időrésben adhat
Szolga: minden páros időrésben adhat

- 1 mesterhez maximum 7 szolga tartozhat egyszerre

Szatellit összeköttetés

Újabb alkalmazási mód. Átjátszóállomás a világűrben, Földről nézve egy pont (Föld felszínétől 36000km távolságra, keringési sebessége "=" a Föld forgási sebességével). Ráküldve az információt, visszasugározza egy másik hullámhosszon. A besugárzott terület hatalmas. Megfelelő erősséggel besugárzott területet talppontnak hívjuk. Ezen a területen belül bárhol tudunk összeköttetést létrehozni. Előnyei: nincs szükség hosszú időre, a kapcsolat pillanatok alatt létrejön, és nem drágább a kábelnél.

4. tétel

A 802.11a/b/g szabványok részletes bemutatása. A különböző szabványokban használt frekvenciák összehasonlítása. A WLAN hálózatok hardverei. WLAN topológiák és felhasználási területeik.

WLAN (Wireless Local Access Network - vezeték nélküli helyi hozzáférési hálózat, vezeték nélküli LAN): Azokat a vezeték nélküli megoldásokat nevezzük így összefoglaló néven, melyek néhány métertôl legfeljebb néhány kilométerig terjedô távolságból lehetôvé teszik, hogy egy LAN hálózat szolgáltatásait igénybe vehessük. A vezeték nélkül elért hálózat lehet egy kiépített vezetékes LAN hálózat, de lehet teljes egészében WLAN hálózat is. Az elérés módja általában nagysebességû rádiófrekvenciás összeköttetés, egyes esetekben pedig az infravörös tartományban mûködô megoldás.

Vezeték nélküli hálózatok WLAN

- 2000 környékén terjedt el széles körben
- Manapság bizonyos esetekben alternatívája a vezetékes hálózatoknak
- Alacsony ár, egyszerű kiépíthetőség
- Mobil felhasználási lehetőségek
- Hatósugár: Épületeken belül akár 100 méter, Épületeken kívül akár 300 méter, tovább növelhető antennákkal, ismétlőkkel.

WLAN szabványok - Home RF: Legkorábbi szabvány

- Működési frekvencia: 2,4 GHz
- Maximális sebesség:1,6 Mbit/s (v 1.2 - 2001-ig), 10 Mbit/s (v 2.0 - 2001 végétől)

WLAN szabványok IEEE 802.11b

- Működési frekvencia: 2,4 GHz
- Maximális sebesség: 11 Mbit/s

WLAN szabványok IEEE 802.11a

Működési frekvencia: 5 GHz
Maximális sebesség: 54 Mbit/s
Nem kompatibilis visszafelé a 802.11b szabvánnyal

WLAN szabványok IEEE 802.11g

Működési frekvencia: 2,4 GHz
Maximális sebesség: 54 Mbit/s
Kompatibilis visszafelé a 802.11b szabvánnyal

WLAN szabványok IEEE 802.11n

Működési frekvencia: 2.4 és/vagy 5 GHz
MIMO (Multiple In/Out) technológia – zavarmentesebb átvitel
Maximális sebesség: 600 MBit/s (MIMO)

WLAN hardver elemek

Rádiófrekvenciás hálózati kártyák

- Csatlakoztatás: PCI, mini PCI, PCMCIA, CF, USB

Hozzáférési pontok

- Rádiófrekvenciás + vezetékes hálózati kártya
- Általában HTTP protokollon keresztül konfigurálható

WLAN Routerek

- Többportos Ethernet router + hozzáférési pont
- Jellemző szolgáltatások: NAT(belső privát hálózat gépeinek IP címeit teljesen elrejthetjük a külső hálózat nyilvánossága elöl), DHCP, Firewall, Repeater

Ismétlők (repeater)

- Hatósugár kiterjesztése jelformázással és erősítéssel
- Összes keret újraküldése azonos csatornán, duplázódó forgalom

Antennák

- Körsugárzó vagy irányított antennák
- Csatlakoztatási lehetőségek az eszközökön
- Előírt kimenő teljesítmény

WLAN topológiák

Ad-hoc hálózatok

- két vagy több kliens összekapcsolása egymással, nincs kiemelt elem

Menedzselt vagy infrastrukturális hálózatok

- a kliensek egy kiemelt elemen (AP) keresztül kapcsolódnak
- A forgalom szűk keresztmetszete lehet az AP
- Lehetőség van hitelesítésre, forgalom szűrésre, a hozzáférések kontrollálására

5. tétel

Rétegszemlélet. A rétegekre bontás okai, a rétegekkel szemben támasztott követelmények. A szolgálatok és szolgálatprimitívek szerepe. Az OSI hivatkozási modell átfogó, általános bemutatása.

Az OSI modell

Az ISO (International Standards Organization) szabványügyi szervezet kiadott egy szabványt a számítógépes hálózatokra, melynek neve OSI (Open System Interconnection - Nyílt rendszerek összekapcsolása).
Az OSI szabványosította a hálózati funkciók rétegbeosztásait:

- Szabványosította a rétegközi *interfészeket*.
- Szabványosította a rétegközi *protokollokat*.
- Viszont nem szabványosította, hogy az egyes rétegek a feladataikat hogyan végezzék el.

Az ISO túlságosan későn hozta létre az OSI szabványt, amire szabványosították a hálózati kommunikációt, addigra már léteztek jól működő számítógép hálózatok, ezért OSI modell szerint épült hálózat nincs. Sem az OSI interfészek, sem az OSI protokollok nem terjedtek el. Az egyedüli tulajdonság ami a mai hálózatokat is jellemzi, az a szalámi-elv vagy rétegstruktúra.

Az OSI modell rétegeinek feladatai

Az eredeti OSI modell 7 rétegű volt.

7	**Alkalmazási réteg**
6	**Megjelenítési réteg**
5	**Viszonylati réteg**
4	**Szállítási réteg**
3	**Hálózati réteg**
2	**Adatkapcsolati réteg**
1	**Fizikai réteg**

Fizikai réteg: Feladata a bitek továbbítása a kommunikációs csatornán olyan módon, hogy az adó oldali bitet a vevő is helyesen értelmezze (a 0-át 0-nak, az 1-et, 1-nek). Kérdések: a fizikai közeg, és az információ tényleges megjelenési formája, egy bit átvitelének időtartama, egy vagy kétirányú a kapcsolat, hogyan épüljön fel egy kapcsolat és hogyan szűnjön meg, milyen legyen az alkalmazott csatlakozó fizikai, mechanikai kialakítása?

Adatkapcsolati réteg: Feladata adatok megbízható továbbítása az adó és fogadó között. Az adatokat **adatkeretekké (data frame)** tördeli, ellátja kiegészítő cím, egyéb és ellenőrző információval, ezeket továbbítja, majd a vevő által visszaküldött nyugtakereteket (acknowledgement frame) véve ezeket feldolgozza. Felmerülő problémák: hogyan jelezzük a keretek kezdetét és a végét, mi történjék akkor ha egy keret elvész, mi történjék akkor ha a nyugtakeret vész el, mi legyen akkor, ha az adó adási sebessége jelentősen nagyobb, mint a vevőké?

Hálózati réteg: Az alhálózat működését biztosítja. A legfontosabb kérdés itt az, hogy milyen útvonalon kell a csomagokat a forrásállomástól a célállomásig eljuttatni.

Szállítási réteg: Feladata a hosztok közötti átvitel megvalósítása (itt már végpontok közötti összeköttetésről van szó, ld. ábra). A kapott adatokat szükség esetén kisebb darabokra vágja, átadja a hálózati rétegnek.

Viszony réteg (más néven **együttműködési réteg**): A különböző gépek felhasználói viszonyt létesítenek egymással, például bejelentkezés egy távoli operációs rendszerbe, állománytovábbítás két gép között.

Megjelenítési réteg: Tipikus feladatai: az adatok szabványos módon történő kódolása, tömörítés, titkosítás.

Alkalmazási réteg: Felhasználói programok (e-mail, fájl átvitel, távoli bejelentkezés, stb.).

Hálózati architektúrák céljai
Összekapcsolhatóság
- Eltérő hardver és szoftver elemek → egységes hálózat

Egyszerű implementálhatóság
- A felhasználók igényeit lefedő általános megoldás

Használhatóság
- A felhasználók hatékony kiszolgálása
- A valós megoldások elrejtése a felhasználók elöl

Megbízhatóság
- Hibák felismerése és javíthatósága

Modularitás, bővíthetőség
- Felhasználói vagy technológiai igény esetén

Rétegszemlélet
A teljes architektúra komplex feladatokat lát el
Interfészek a rétegek között
- Adott csomópontban a szomszédos rétegek között
- Több csomópont azonos rétegei között

Előnyök
- Modularitás
- Eltérő hardver és szoftver alkalmazhatósága

Rétegekkel szembeni követelmények
Minden réteg jól elkülönülő önálló feladatot lát el
A rétegek egymástól függetlenek
A rétegek egymásra épülnek
- Minden réteg az alsóbb rétegek információt használja
- Minden réteg a felsőbb rétegek számára nyújt szolgáltatásokat

A hosztok azonos rétegei egymással kommunikálnak
A kommunikáció az interfészeken valósul meg
A kommunikációhoz a protokollokat használják
- protokoll verem (protocol stack)

Szolgálatok
Szolgáltatások csoportosítása
- Összeköttetés alapú
    - Megbízhatatlan kapcsolat (telefon)
    - Megbízható adatfolyam (SSH)
- Összeköttetés mentes
    - Nem megbízható datagrammok (UDP)
    - Megbízható (nyugtázott) datagrammok (TCP)
    - Kérés - válasz alapú szolgáltatások (adatbázisok)

Leírja, hogy milyen műveletek hajthatók végre
- Nem írja le az implementáció módját → protokoll

Szolgálatprimitívek
Szolgálat típusok
- Megerősített (confirmed)
- Megerősítetlen (unconfirmed)

Osztályok
- Megerősített
    - Listen (blokkolt várakozás bejövő kapcsolatfelvételre)
    - Connect (összeköttetés létrehozása egy várakozó entitással)
    - Receive (blokkolt várakozás bejövő üzenetre)
    - Send (üzenet küldése)
    - Disconnect (összeköttetés bontása)
- Megerősítetlen
    - Receive (blokkolt várakozás bejövő üzenetre)
    - Send (üzenet küldése)

6. Tétel

A TCP/IP hivatkozási modell bemutatása. Az OSI és a TCP/IP modell értékelése és összehasonlítása. A siker vagy sikertelenség okainak elemzése.

Bevezető: Habár az OSI modell általánosan elfogadottá vált, az **Internet nyílt szabványa** történeti és technikai okokból mégis **a *TCP/IP referenciamodell* és a *TCP/IP protokollkészlet* lett.** A *TCP/IP* a világ bármely két pontján (vagy azon kívül) levő számítógépek között biztosít adatkommunikációt. A *TCP/IP* modellt az Amerikai Védelmi Minisztérium definiálta, mert egy olyan hálózatot kívánt létrehozni, amely minden körülmények között – még egy atomháború esetén is – működőképes marad. Képzeljünk el egy háborús övezetet, amelyet keresztül-kasul behálózzák a különböző típusú kommunikációs összeköttetések: vezetékek, mikrohullámú csatornák, optikai szálak és műholdas csatornák! Ilyen helyzetben is szükségünk van információ- illetve adatáramlásra (*csomagok* formájában), függetlenül az egyes csomópontok vagy hálózatok állapotától (amik akár meg is semmisülhetnek a harcokban). Az Amerikai Védelmi Minisztérium azt akarta, hogy a *csomagok* mindenkor, minden körülmények között, bármely pontból bármely pontba eljussanak. Ez egy igen nehéz tervezési probléma, de ez vezetett a *TCP/IP* modell megalkotásához, ami azóta az Internet szabványává vált.

TCP/IP rétegei: A *TCP/IP* modell négy réteget tartalmaz: az *alkalmazási réteget*, a *szállítási réteget*, az *Internet réteget* és a *hálózati réteget*.

Alkalmazási réteg (ez a legfelső) A magas szintű protokollok feladatait is tartalmazta, vagyis a megjelenítést, a kódolást és a párbeszéd-szabályozást. A *TCP/IP* minden alkalmazás szintű feladatot egy rétegbe foglal bele.

Szállítási réteg A megbízhatósággal, az adatfolyam-vezérléssel és a hibajavítással foglalkozik. Az egyik ide tartozó protokoll, a *Transmission Control Protocol (*TCP*)* igen hatékony és rugalmas módon teszi lehetővé a megbízható, gyors, alacsony hibaarányú hálózati kommunikációt. A *TCP* egy kapcsolatorientált protokoll. Ez a protokoll az információkat szegmensekbe csomagolva a forrás- és a célállomás között párbeszédszerű kommunikációt tesz lehetővé.

Internet réteg

Az *Internet réteg* feladata az, hogy az összekapcsolt hálózatok bármely részhálózatában levő forrásállomás *csomagjait* elküldje, és azokat a célállomáson fogadja. Ennek a rétegnek a feladatát az *Internet Protocol (IP)* látja el. A legjobb útvonal kiválasztása és a csomagkapcsolás ebben a rétegben történik. Hasonlítsuk össze ezt a réteget a postai szolgálattal! Amikor feladunk egy levelet, nem tudjuk, az hogyan fog eljutni a címzetthez csak azt, hogy meg fog érkezni.

Hálózati réteg

Ez a réteg foglalkozik az összes kérdéssel, ami ahhoz szükséges, hogy egy *IP-csomag* különböző fizikai összeköttetéseken haladjon keresztül. Ide tartozik az OSI modell fizikai és adatkapcsolati rétegének minden részlete.

A TCP/IP protokolljai:

Alkalmazási réteg protokolljai: fájlátviteli protokoll (**FTP**), hipertext átviteli protokoll (**HTTP**), egyszerű levéltovábbító protokoll (**SMTP**), körzeti névkezelő rendszer (**DNS**)

Szállítási réteg protokolljai: TCP, UDP Hálózati réteg protokollja: IP

Az OSI modell és a TCP/IP modell rétegenkénti összehasonlítása:

Hasonlóságok

mindkettőben található egy alkalmazási réteg, bár funkciójuk igencsak különböző
mindkettő hasonló funkciójú szállítási és hálózati réteggel rendelkezik
csomagkapcsolt (nem pedig áramkörkapcsolt) technológiát vesznek alapul

Különbségek

A TCP/IP az alkalmazási rétegre hárítja a megjelenítési és a viszonyréteg funkcióit
A TCP/IP az OSI modell adatkapcsolati rétegét és a fizikai réteget egy réteggé vonja össze
A TCP/IP kevesebb rétege miatt egyszerűbbnek tűnik

- A TCP/IP protokolljaira épült az Internet, tehát a TCP/IP modell csak a protokolljai miatt nyert létjogosultságot. Ezzel szemben az OSI modellre épülő protokollokat egyetlen hálózat sem használja, bár mindenki az OSI modell alapján gondolkodik.

Az OSI modell

Világszerte elismert, általános, protokoll független szabvány.
Részletesebb, ezért alkalmasabb oktatási célokra.
Részletesebb, ezért jobban használható hibakeresésre.

Ezzel szemben a TCP/IP modell protokolljai az Internet szabványos protokolljai.

7. tétel

A fizikai réteg feladatainak felsorolása. A rétegben használt eszközök (repeater, HUB, média konverter) feladatainak és működésének jellemzése. A szinkron és az aszinkron átvitel. Jelzésátvitel.

1. réteg: A fizikai réteg

A *fizikai réteg* írja elő a végrendszerek közti fizikai összeköttetések kialakításának, fenntartásának és lebontásának elektromos, mechanikus és funkcionális követelményeit. A fizikai réteg definiálja a **feszültségszinteket**, a **feszültségváltozás időzítését**, a **fizikai adatsebességet, a maximális átviteli távolságot, a csatlakozókat** és más hasonló paramétereket. A fizikai réteg az adatkapcsolati rétegnek biztosít szolgáltatásokat. A fizikai réteg az adatkapcsolati réteg *kereteit* egyesek és nullák (bitek) sorozatára kódolja, és továbbítja az 1. rétegbeli átviteli közegen (ami rendszerint egy vezeték).

Átviteli közeg azt a csatornát jelöli, amin az adatok áthaladnak. Ez az alábbiak közül bármelyik lehet: telefonvezeték, UTP kábel, koaxális kábel, optikai szál, más típusú rézkábel.
Az átviteli közeg az 1. réteg alatt helyezkedik el, vagyis nem része az 1. rétegnek, noha a kommunikáció fizikailag ezen keresztül zajlik.
A hálózati kommunikációban másik, kevésbé nyilvánvaló *átviteli közeg* is használatos, a levegő amin közvetítik a rádióhullámokat, a mikrohullámokat. Az olyan kommunikációt, amihez nincs szükség semmilyen vezetékre vagy kábelre, *vezeték nélküli* kommunikációnak nevezzük.

Szinkron átvitel jellemzői: A bitszervezésű üzenetek jellegzetes átvitelmódja. Szinkron kommunikáció esetén a két kapcsolatba lépő állomás egyezteti az adatátviteli sebességet, vagyis azt, hogy másodpercenként hány darab jel kerüljön továbbításra, valamint meddig történhet az adás. Az adó elkezdi az adást, és a rendelkezésre álló idő alatt átküldi a továbbítandó üzenetnek bitjeit. A vevő folyamatosan veszi a biteket és összeállítja azokat számára feldolgozható byte-okká.
Aszinkron átvitel jellemzői: Nincs folyamatos adatátvitel, nincs közös szinkron, az adó és a vevő összhangban kell, hogy működjön. Az átviteli hibák jelzése paritás bitekkel történik.

Eszközök

Hub: a számítógépes hálózatok egy hardvereleme, amely fizikailag összefogja a hálózati kapcsolatokat. Másképpen szólva a hub a hálózati szegmensek egy csoportját egy hálózati szegmensbe vonja össze, egyetlen ütközési tartományként láttatja a hálózat számára. Leegyszerűsítve: az egyik csatlakozóján érkező adatokat továbbítja az összes többi csatlakozója felé. Ez passzívan megy végbe, anélkül, hogy ténylegesen változtatna a rajta áthaladó adatforgalmon. A repeatertől eltérően jelerősítést nem végez.
A hubok között 2 alaptípust különböztetünk meg:
* aktív hub: az állomások összefogásán kívül a jeleket is újragenerálja, erősíti.
* passzív hub: csupán fizikai összekötő pontként szolgál, nem módosítja vagy figyeli a rajta keresztülhaladó forgalmat.

A legelterjedtebbek a 8, 16, 24 portos eszközök, de találkozhatunk kisebb, 4 portossal is. A passzív hubok elektromos tápellátást nem igényelnek. Az intelligens hubok aktív hubként üzemelnek, mikroprocesszorral és hibakereső képességekkel rendelkeznek.

Repeater: ismétlőt, jelismétlő, a jelek újragenerálására használt hálózati készülék. Újragenerálja az átvitel közbeni csillapítás miatt eltorzult analóg v. digitális jeleket. Az ismétlő nem végez intelligens forgalomirányítást.
Média konverter: Különböző átviteli közegek illesztésére szolgál pl: Csavart érpár -> optikai szál, Soros vonal -> optikai szál.

Jelzésátvitel (hol továbbítjuk a vevő felé a különböző jelzéseket?)
Sávon belüli jelzésátvitel
A hasznos adatok és jelzések közös sávban
Fenntartott sávok a jelzéseknek
Átviteli problémák adódhatnak ebből
Sávon kívüli jelzésátvitel

8. Tétel

Az adatkapcsolati réteg feladatainak felsorolása. Nyugtázott, nyugtázatlan szolgálatok. A keretezés feladata és megvalósítása az adatkapcsolati rétegben. A lehetséges megoldások bemutatása példák segítségével.

Adatkapcsolati réteg: Nyers bináris adatokat továbbít a fizikai és a hálózati szint között. Az adatkapcsolati réteg három feladatot hajt végre: a hálózati rétegektől kapott információkat keretekbe rendezi, hibavédelmet valósít meg, valamint a keretek továbbítását és nyugtázását végzi.

Nyugtázatlan összeköttetés mentes szolgálat

Egymástól független keretek küldése a forrástól a célig
Nincs előzetes kapcsolatépítés és bontás
Az elveszett kereteket nem állítja helyre a réteg
Alkalmazhatóság: Alacsony hibaarány esetén (javítás a felsőbb rétegekben), Valós idejű adatforgalom esetén (hangátvitel)

Nyugtázott összeköttetés mentes szolgálat

Szintén nincs előzetes kapcsolatépítés és bontás
Minden egyes keret megérkezését nyugtázza a cél állomás
Alkalmazhatóság Pl.: vezeték nélküli kapcsolatoknál

Nyugtázott összeköttetés alapú szolgálat

Összeköttetés felépítése az adatátvitel előtt
Sorszámozott keretek
- minden keret garantáltan megérkezik
- minden keret garantáltan egyszer érkezik meg
- minden keret a megfelelő sorrendben érkezik meg

Az átvitel folyamata:
- összeköttetés felépítése (számlálók, változók inicializálása)
- keret(ek) továbbítása
- összeköttetés bontása (számlálók, változók felszabadítása)

Keretezés:

Nagyszerű technikai vívmány a fizikai átviteli közegen áramló bitek kódolása, de az önmagában nem elég a kommunikációhoz. A keretezés révén olyan információk is továbbíthatók, illetve kiolvashatók, amelyeket a bitfolyamok kódolása önmagában nem tesz lehetővé:

– mely számítógépek kommunikálnak egymással
- az egyes számítógépek közti kommunikáció mikor kezdődik, és mikor fejeződik be
- a kommunikáció során bekövetkezett hibák jegyzéke

A keret az adat kezdetét és végét jelöli, így megkönnyíti az adat továbbítását. Emellett a keret megóvja az adatokat a sérüléstől.

Egy általános keret felépítése.

Nagyon sokféle kerettípus létezik, ezek részleteit a megfelelő szabványok írják le. Egy általános keret mezőkből áll, melyek bájtokat tartalmaznak. A mezőnevek a következők:

- keretkezdő mező
- címmező
- hossz/típus/vezérlő mező
- adatmező
- keretellenőrző mező
- keretlezáró mező

Minden keret egy keretkezdő, jelző bájtsorozattal kezdődik. Minden keret tartalmaz címinformációt, például a forrásszámítógép MAC-címét és a célszámítógép MAC-címét. Minden keretben vannak speciális mezők. Egyes technológiáknál használják a "hossz" mezőt, mely megadja a keret pontos hosszát. Más keretekben "típus" mező is található, mely a harmadik rétegbeli küldő protokollt adja meg. A rendszer esetenként kitöltőbájtokkal egészíti ki a kereteket annak érdekében, hogy azok hossza elérje a minimális értéket (amire időzítési okokból van szükség). A keret végét jelezheti a keret hossza is, ekkor a keretellenőrző mező (FCS) után a keret véget ér. Olyan megoldás is lehetséges, hogy a keret végét egy speciális bitsorozat, a végjelző mutatja.

9. Tétel

Hibajelzés és javítás az adatkapcsolati rétegben. Egybites és csoportos hibák fogalma, jelzése, javítása. A CRC hibaellenőrzés erősségei és gyengeségei. A hibajavító Hamming-kód és a CRC működésének bemutatása egy-egy példa segítségével.

Az adatkapcsolati réteg nyers bináris adatokat továbbít a fizikai és a hálózati szint között. Alapvető feladata, hogy egy hibától mentes adatátviteli vonalat biztosítson, amelyen az adatok meghibásodás nélkül eljutnak a hálózati réteghez. Ezt úgy valósítja meg, hogy a küldő fél a bemenő bináris adatokat keretekké (frames) tördeli a hálózati szint számára, a kereteket sorrendhelyesen továbbítja, majd a vevő által visszaküldött, az átvitelt igazoló nyugtakereteket feldolgozza. Mindegyik keret egy ellenőrző összeggel van ellátva. A keret megérkezése után ez az ellenőrző összeg a vételi oldalon a vett adatokból is kiszámításra kerül. Ha ez az összeg nem egyezik meg a kiinduló összeggel, akkor a keretet a vevő eldobja, és az adónak meg kell ismételnie a keret elküldését. Az adatkapcsolati szint a hálózati szintről is fogad adatokat, amelyeket hibátlan bináris formátummá alakít át az alatta levő fizikai réteg számára. Mivel a fizikai réteg csupán a bitfolyam adásával, valamint vételével foglalkozik, ennek a rétegnek a feladata az adatkeret határok létrehozása, felismerése. Ezt speciális bitmintáknak a keretek elé, illetve mögé helyezésével éri el.

Problémák lehetnek

- Megérkezett-e minden keret
- Minden keret csak 1-szer érkezett-e meg
- Jó sorrendben érkeztek-e meg a keretek

Megoldások:

- Nyugták alkalmazása melyekből kiderül hogy megérkeztek-e a keretek
- Kimenő keretek sorszámozása ebből kiderül, hogy jó sorrendben érkeztek-e meg illetve, hogy nincs-e 2 valamelyikből.
- Időzítők használata: ez arra kell, hogy ha nem jön nyugta, időben akkor újra küldi a csomagot mert biztos elveszett a csomag. Az is lehet hogy a nyugta veszik el de újraküldi a küldő így 2-szer fog megérkezni a vevőhöz.

Egybites és csoportos hibák

- **Egybites hibák**
- **Csoportos hibák**
  - Bizonyos közegek esetén sokkal gyakoribb
  - Azonos hibaarány mellett kevesebb hibás keret
  - Jelzése és javítása lényegesen nehezebb

Hibakereső, hibajavító eljárások

1. Ismétlő módszerek

Az elküldött adatok bitsorozatát bitek blokkjaira tördelik, és küldésnél minden blokkot egy előre megadott számszor újraküldenek. Például, úgy küldjük el a "1011" bitsorozatot, hogy minden bitnégyest háromszor küldünk el.

Tegyük fel, hogy elküldtük a "1011 1011 1011" sorozatot, amit "1010 1011 1011" sorozatnak veszünk. Mivel egy csoport eltér a másik kettőtől, megállapítható, hogy az átvitel hibás volt. Ez a megoldás nem túl hatékony.

2. Paritásbit

Az adatok végére egyetlen paritásbitet függesztenek. A paritásbitet úgy választják meg, hogy 1-esek száma a kódszóban páros (vagy páratlan) legyen. Mivel minden egyes bithiba rossz paritású kódszót hoz létre, ezért csak egybites hibák észlelésére alkalmas.

3. Hamming-távolság

=Két kódszó bitpozícióban mért távolsága: Hamming-távolság (pl.:10001001, 10110001-nél 3, mert 3 bitben különböznek). **KIZÁRÓ VAGY kapcsolattal lehet megnézni,** mennyi. Ha két kódszó d Hamming – távolságra van egymástól, az egyik a másikba d db egybites hibával mehet át. **d hiba észleléséhez d+1 H-távolság kell**

(ekkor d hiba még nem hoz létre új kódszót). d hiba javításához 2d+1 H-távolság kell, mert d bit változása esetén is a hibás kódszó az eredeti kódszóhoz közelebb áll.

Tehát gyakorlatban: Vannak megadott kódszavaink amiknek a Hamming távolsága fix pl:3. Ekkor ha jön 1 olyan szó ami valamelyiktől 1-el tér csak el, akkor az biztosan az volt eredetileg tehát javítható a hiba.

4. A CRC (ciklikus redundancia ellenőrzés) az adatblokkot egy polinom együtthatóinak tekinti, amelyet eloszt egy előre meghatározott, állandó polinommal. Az osztás eredményének együtthatói alkotják a redundáns adatbiteket, a CRC-t.

A vett adat ellenőrzéséhez elegendő megszorozni a CRC-t az előre meghatározott polinommal. Ha a szorzás eredménye a hasznos adat, akkor az adatok hiba nélkül érkeztek meg.

Másik lehetséges ellenőrzési mód a CRC újbóli kiszámítása a hasznos bitekből, és a vett CRC-vel való összehasonlítása.

Előnyei: Minden 1 bites hiba észlelhető, akár két izolált 1 bites hiba is. A csoportos hibák is jelezhetőek.

Hátránya: Bonyolultnak tűnik, bár egyszerűen megvalósítható logikai áramkörökkel.

10. tétel

Statikus és dinamikus csatornakiosztás a közegelérési alrétegben. ALOHA és réselt ALOHA. A CSMA és CSMA/CD jellemzése. A Manchester kódolás bemutatása egy példán keresztül. Ethernet hálózatok jellemző átviteli közegei.

Statikus csatornakiosztás: a csatornát több alcsatornára osztjuk fel (célszerűen az egyidejű felhasználók számától függően). Az alcsatornákat fixen hozzárendeljük egy összeköttetéshez, és nem vesszük figyelembe sem az alcsatorna pillanatnyi állapotát, sem az adatátviteli igényt. Ha egy összeköttetés nem akar semmit továbbítani, akkor a hozzárendelt alcsatorna kihasználatlan marad.
Klasszikus módszere az FDM (frekvenciaosztás). Például a rádiók osztoznak a rendelkezésre álló sávon. Minden állomás eltérő, és előre rögzített frekvencián ad, ami lehetővé teszi az állomások szétválasztását a bemeneten
Dinamikus kiosztás: figyelembe vehetjük a pillanatnyi terhelési állapotokat, és a rendelkezésre álló szabad útvonalakat is.

ALOHA protokoll: Felhasználók versengése a közös csatornáért, alapgondolata egyszerű: mindenki akkor ad, mikor akar, és a sikertelen kereteket megismételjük. A sikertelen keretek azonnali újraadása újabb ütközéshez vezetne, ezért késleltetjük az adást.
- Réselt ALOHA: képes arra, hogy egy egyszerű ALOHA rendszer kapacitását megduplázzák. A módszer lényege, hogy az időt szeletekre osztják, oly módon hogy minden felhasználónál ugyanazok az időintervallum határok pontos helyei - ezt egy szinkronjel segítségével határozzák meg.

CSMA/CD: Közös kommunikációs csatorna több fél közötti elosztásának módját és szabályait meghatározó rendszer. Szabályrendszere biztosítja, hogy a közös csatornán kommunikáló felek képesek legyenek a szimultán adások és az ezekből következő ütközések detektálására, valamint ezek elhárítására.
Amennyiben több fél azonos idejű adási kísérlete miatt ütközés jön létre, úgy az érintett felek külön-külön, de véletlenszerűen meghatározott ideig felfüggesztik adási kísérleteiket abban a reményben, hogy így a következő próbálkozás alkalmával már nem egyszerre próbálnak majd meg adni. Amennyiben az újabb adási kísérlet során ismét ütközés jön létre, úgy az adni kívánó felek a korábbiaknál egyre tovább várnak ki - így csökkentve az újabb ütközések esélyét. Az egyik legismertebb CSMA/CD eljárást alkalmazó architektúra az Ethernet.

Ethernet
Az Ethernet adatszórásos átviteli közeg. Ez azt jelenti, hogy a hálózat minden készüléke látja az átviteli közegen átmenő összes adatot. Mindazonáltal nem minden készülék dolgozza fel az adatot, csak az a készülék másolja le az adatot, melynek MAC- és IP-címe megegyezik az adatban tárolt MAC- és IP-címmel. Miután a célkészülék ellenőrizte az adatban tárolt MAC- és IP-címet, a készülék törli a hibásnak talált adatcsomagokat. A célkészülék nem értesíti a küldő készüléket sem arról, ha az adatcsomag sikeresen megérkezett, sem arról, ha nem. Az Ethernet egy összeköttetés-mentes hálózati architektúra, amelyre "leghatékonyabb kézbesítésre törekvő rendszerként" (best-effort delivery system) is szoktak hivatkozni.

Átviteli közegei
UTP - csavart vagy sodrott érpár. A kábelek általában négy csavart érpárt tartalmaznak. Átviteli zavar csökkenthető árnyékolással (STP).
További közegek: koaxiális kábel, optikai szálak.

Manchester-kódolás: A vezeték nélküli kommunikációhoz, fejlesztették ki.
A Manchester kódolósnál egy bájt átviteléhez két bájtot kell küldenünk. A következőképpen néz ki:
- A logikai 1-es szintet átalakítjuk 10-re. (nem tíz, hanem egy és nulla)
- A logikai 0-ás szint pedig a 01 lesz.
Pl. 11100011 Manchester-kódja: 10 10 10 01 01 01 10 10
Így láthatjuk, hogy a kódolt adatnál minden bitet két bittel írunk le, és nincsen folyamatos jel. Mindig csak pillanatokra kapcsoljuk ki-be.

11. tétel

A DIX és az IEEE 802.3 keretformátum részletes bemutatása és összehasonlítása. Az Ethernet ütközéskezelési technikája. Kapcsolás Ethernet hálózatokban.

Az **Ethernetet** eredetileg arra tervezték, hogy lehetővé tegye két vagy több állomás számára ugyanazon átvitel közeg használatát úgy, hogy a jelek között ne keletkezzen interferencia.

Az Ethernet sikere a következő tényezőknek köszönhető:

- Egyszerűség és könnyű karbantartás
- Az új technológiák átvételének képessége
- Megbízhatóság
- Alacsony telepítési és bővítési költségek

Az eredeti Ethernetet "Kísérleti Ethernet"nek nevezik ma. Kevés helyen használták, de a gondolat többek fejében szeget ütött. Az első "Ethernet", amit a Xerox-on kívül használtak, a DIX Ethernet volt. (DEC – Intel – Xerox egyesülés)

DIX és IEEE keretezése.

A keretszerkezet gyakorlatilag az Ethernet összes változatánál azonos, a 10 Mbit/s sebességűtől kezdve egészen a 10 000 Mbit/s sebességűig.

A **DIX** *(ez volt korábban)* által kifejlesztett Ethernet-változatnál **az előtag és a keretkezdet jelző egyetlen mezőben volt.** Maga a bináris bitsorozat azonos volt. **A hossz/típus mező** a korai IEEE-változatokban hosszként szerepelt, míg a **DIX-változatban típusnak hívták.** A későbbi IEEE-változatokban a két mezőt hivatalosan is egyesítették, ugyanis mindkét használati mód gyakori volt.

A 802.3 Ethernet keretekben a következő mezők szerepelhetnek. A mezők egy részének kötelező meglennie, egy részük pedig elhagyható: Előtag, Kezdetjelző, Célcím, Forráscím, Hossz/típus, Fejrész és adat, Keretellenőrző összeg, Kiterjesztés

Az előtag váltakozva tartalmaz egyeseket és nullákat. A 10 Mbit/s-os és kisebb sebességű Ethernet-megvalósításoknál az órajel szinkronizálása ennek a mezőnek a segítségével történik. Az Ethernet gyorsabb változatai szinkron működésűek, ezeknél időzítési információkra nincs szükség; ennek ellenére, a kompatibilitás érdekében a mező megmaradt.

A kezdetjelző egy egyoktettes mező, amely az időzítési információk végét jelzi. Tartalma egy bitsorozat.

A célcím lehet egyedi, csoportos vagy szórásos.

A forráscím a forrás MAC-címet tartalmazza.

A hossz/típus mezőt kétféle célra lehet használni. Ha értéke a decimális 1536-nál,
kisebb, akkor a benne szereplő érték hosszt ad meg. A típus érték azt adja meg, hogy az Ethernet folyamatainak lezárulása után melyik felsőbb rétegbeli protokoll fogja kapni az adatokat.

Az adat mező és a szükség szerinti kitöltés hossza tetszőleges lehet, feltéve, hogy a keret
mérete nem haladja meg a felső mérethatárt. A maximális átviteli egység az Ethernet esetében
1500 oktett, az adatok mérete tehát ezt nem haladhatja meg. Ha nincs elég adat ahhoz, hogy a keret mérete elérje a minimális kerethosszt, akkor előre meg nem határozott mennyiségű adat kerül beillesztésre, közvetlenül a felhasználói adatok mögé. Ezt a többletadatot nevezzük kitöltésnek. Az Ethernet keretek hosszának 64 és 1518 oktett között kell lennie.

Az FCS mezőben egy 4 bájtos CRC érték szerepel, amelyet az adatokat elküldő készülék számít ki, majd a célkészülék a hibás keretek felismerése céljából szintén meghatároz.

Az ETHERNET ütközéskezelési teknikája

Ütközések kezelése
Ütközések után véletlen idej- várakozás
Kettes exponenciális visszalépés
_ 1. ütközés: 0 vagy 1 id_résnyi várakozás
_ 2. ütközés: 0, 1, 2 vagy 3 id_résnyi várakozás
_ 3. ütközés: 0 … 7 id_résnyi várakozás
_ n. ütközés: 0 - 2n-1 id_résnyi várakozás
_ Maximális intervallum a 10. ütközés után 0 … 1023
_ Hibaüzenet a 16. ütközés után
Miért nem választunk azonos számú lehetőségből?
_ Sok állomás együttes ütközése
_ Néhány állomás ütközése

Kapcsolási módok

- **Közvetlen kapcsolás**
  - A MAC célcím megérkezése után kezdődik a továbbítás
  - Minimális kapcsolási késleltetés
  - Hibaellenőrzésre nincs lehetőség
  - Csak szimmetrikus kapcsolás valósítható meg

- **Töredékmentes továbbítás**
- Az első 64 bájt után kezdődik a továbbítás
- Ellenőrizhető a címek és a protokollinformációk helyessége
- **Tárol és továbbít módszer**
- A keret továbbítása csak a teljes keret vétele után történik
- Újraszámolható a keret ellenőrző összege
- Egy hibás keret nem kerül továbbításra, azonnal eldobható
- Aszimmetrikus kapcsolás is megvalósítható

12. Tétel

Sebesség és hatékonyság az Ethernet hálózatokban. A szórási és ütközési tartományok szerepe, jelentősége, kialakítása a gyakorlatban. Üzenetszórás a második rétegben. Szórási vihar.

Az Ethernet számos kábeltípuson (koax, réz stb.) működik 1978 óta legalább 10 Mbps sebességgel, de azóta is fejlődik, létezik már a 100 Mb/s-os és még nagyobb sebességű változat is, a kábelezés is változott.
Ahhoz, hogy a lehető legnagyobb sávszélességű és teljesítményű LAN-t lehessen kialakítani, a következő szempontokat kell figyelembe venni a LAN megtervezése során:
- A kiszolgálók funkciója és elhelyezése
- Az ütközési tartományok kérdésköre
- A szegmentálás kérdésköre
- A szórási tartományok kérdésköre

Ütközési tartomány: olyan fizikai hálózati szegmens, ahol az azonos vivő közegbe küldött adatcsomagok egymással „ütközni" képesek. Az ütközés olyan esemény, melynek során egy hálózati eszköz elküld egy csomagot a hálózati szegmensen, amivel az azonos szegmensen lévő minden más eszköznek is foglalkoznia kell. Eközben, egy másik eszköz ugyanezt teszi, és az egymással versenyző csomagok eldobásra kerülnek, majd újraküldik őket. Ez a jelenség a hálózat hatékonyságát csökkenti.
Szórási tartomány: a készülékeknek az a halmaza, amelynek elemei a halmazba tartozó minden más készülék szórásos keretét megkapják. Minden állomásnak, amely szórásos keretet kap, fel kell azt dolgoznia. Ezek feldolgozása erőforrásokat emészt fel, csökkenti az állomás rendelkezésre álló sávszélességét.

Szegmentálásnak azt a folyamatot nevezzük, amikor egy ütközési tartományt több kisebb ütközési tartományokra osztunk fel. Javasolt hálózati kapcsoló (switch) használata, ami megnöveli az ütközési tartományok számát, de drasztikusan csökkenti azok méretét. Egy switch minden portja saját ütközési tartományt alkot.
A forgalomirányítóval (router) történő szegmentálás további előnye, hogy a szórási tartomány méretét is csökkentik, valamint ki tudják választani a legjobb útvonalat.

Csoportos címzés: Több állomás elérése egyetlen csoportcímmel.
Vannak olyan információk amikről a hálózat minden hosztjának tudnia kell. Ekkor küldenek a gépek csoportos címzéssel adatot. (pl. forgalomirányító, amikor bekapcsol, megkérdez dolgokat a többitől és ezt szórással teszi vagy elküld információkat magáról a hálózatba) Viszont figyelni kell, hogy ne legyen túl sok szórás a hálózaton.

Szórási vihar
- A szórási és csoportcímzéses forgalom telíti a hálózatot
- Újabb kapcsolatok nem hozhatók létre
- A meglévő kapcsolatok megszakadhatnak
- Szélsőséges esetben leállhat a hálózati forgalom

A második réteg eszközei továbbítják a szórást
- Szórási tartományok létrehozása
- Harmadik rétegbeli eszközökkel (router)
- A harmadik rétegbeli működés teszi lehetővé a szórási tartományok szegmentálását
- MAC címek helyett IP címek használata (3. réteg)

13. Tétel

A 802.3u és a 802.3z Ethernet szabványok kialakulása, működése, felhasználása. A lehetséges átviteli közegek és eszközök jellemzése.

802.3u (Fast Ethernet)

A 10MBit/s-os Ethernet hálózat sebessége a 90-es évekre kezdett kevésnek bizonyulni, ezért fejlesztették ki a **Fast Ethernet**et a maga 100MBit/s-os átviteli sebességével. Két fajtája létezik, a **100BASE-TX**, amely rézalapú UTP átviteli közeget használ, illetve a **100BASE- FX,** amely többmódusú optikai szálat alkalmaz. A 100 Mbit/s sebességű változatok keretformátuma a kompatibilitás miatt a 10 Mbit/s sebességűekével azonos. Mindkét változat azonos időzítéseket, keretformátumot és átvitelt használ. A nagyobb frekvenciájú jelek érzékenyebbek a zajokra.
A TX fél-duplex módban 100 Mbit/s, duplex módban pedig 200 Mbit/s sebességű adatátvitelre képes. A 100BASE-TX összeköttetések ismétlő nélkül legfeljebb 100 méteres távolság áthidalására alkalmasak, kapcsolókkal ez kitolható.
Az optikai szálas (FX) változatra elsősorban emeletek és épületek közötti összeköttetésekhez kellett, ahol a rézkábelek használata előnytelen volt; továbbá erősen zajos területekhez. Soha nem vált sikeressé azonban a réz- és optikai kábel alapú Gigabit Ethernet megjelenése miatt.

802.3z (Gigabit Ethernet)

A gyors Ethernet szabványt követően 1995-ben a 802-es bizottság egy még gyorsabb Ethernet tervein kezdett dolgozni. A célkitűzések a következők voltak: 10x gyorsabb sebesség, kompatibilitás az eddigi Ethernetekkel. A végső szabvány, a 802.3z eleget tett a feltételeknek.

A gigabites Ethernet – eltérően a klasszikus Ethernettől – pont-pont felépítésű. A legegyszerűbb topológiánál a két számítógép van gigabites Ethernettel összekapcsolva. Gyakoribb az a megoldás, amikor egy kapcsoló vagy elosztó köt össze több számítógépet, vagy további elosztókat vagy további kapcsolókat. Minden esetben egy Ethernet kábel végén pontosan egy-egy eszköz található csak.

Az **1000BASE-T** a rézkábel alapú verzió. Ennél mivel a bitek rövidebb ideig, gyorsabban kerülnek rá az átviteli közegre, az időzítések kritikus szerepet játszanak. A nagysebességű átvitelekhez magasabb frekvenciákat kell használni, emiatt viszont a rézkábeleken utazó bitek érzékenyebbek a zajokra. Az UTP kábelek megegyeznek a 10BASE-T/TX kábelekkel, kivéve azt, hogy az összeköttetéseknek jobb minőségűeknek kell lenniük. Az adatok küldése mindkét irányban egyszerre, ugyanazon a vezetéken folyik. Ez folyamatos ütközéseket eredményez az érpárokon. Az 1000BASE-T a fél-duplex és a duplex működést egyaránt támogatja, ám a duplex az elterjedtebb.

Az **1000BASE-X** szabvány az optikai szálakon végzett, 1 Gbit/s sebességű, duplex átvitelt definiálja. Ez a **8B/10B** kódolást használja. Ezt az optikai szálon egyszerű nullára vissza nem térő vonali kódolás követi. Mivel a LED vagy a lézerforrás teljes be- és kikapcsolásához idő kell, a fényimpulzusok valójában a fényerő erősödését és gyengülését jelentik. (logikai 0-k alacsonyabb fényerejű, 1-esek pedig erősebb fényű szakaszok)
Egyéb fajták: az **1000BASE-SX** *(rövidhullámú lézerrel /LEDdel)* és az **1000BASE-LX** *(hosszúhullámú lézerrel).*
Ezek azonos időzítési paramétereket használnak. A bitidő mindegyiknél 1ns vagyis a másodperc milliárdod része.

A keretformátum szintén megegyező, szintén kompatibilitás miatt.
A Gigabit Ethernet összeköttetésnél két állomás között csak egyetlen ismétlő lehet.

14. Tétel

Összeköttetés alapú és összeköttetés mentes szolgálatok a hálózati rétegben. Forgalomirányítás adaptív és nem adaptív algoritmusok segítségével. A legrövidebb útvonal és az elárasztás.

Összeköttetés-mentes hálózati szolgáltatások.

A hálózati szolgáltatások nagy része valamilyen *összeköttetés-mentes* kézbesítési rendszert használ. Ez azt jelenti, hogy minden csomagot külön kezelnek, illetve küldenek át a hálózaton. Az egyes csomagok különböző útvonalakon juthatnak el a célállomásig, ahol újra összerakják őket. Az összeköttetés-mentes rendszerekben a csomag elküldése előtt nem veszik fel a kapcsolatot a célállomással. Az összeköttetés-mentes rendszert leginkább a postai rendszerhez hasonlíthatjuk. Mielőtt a levelet az egyik helyről a másikra elküldik, nem veszik fel a kapcsolatot a címzettel. A levelet elküldik, de a címzett csak akkor szerez róla tudomást, amikor azt megkapja. Szükséges teljes cél- és forráscím minden csomaghoz. Vonalszakadás esetén csak néhány adat fog elveszni.

Összeköttetés alapú hálózati szolgáltatások

Az *összeköttetés-alapú* rendszerekben az adatátvitel előtt összeköttetést építenek ki az adó és a vevő között. Az összeköttetés-alapú hálózatokat a telefonrendszerhez hasonlíthatjuk. Először tárcsázunk, ezután létrejön a kapcsolat, s csak ekkor kezdődhet a kommunikáció. Csak a virtuális áramkör azonosítója kell. Hiba esetén pl vonalszakadás az összes virtuális áramkör megszakad, ezért sok adat elveszik.

Az összeköttetés-mentes és az összeköttetés alapú hálózati folyamatok összehasonlítása.

Az összeköttetés-mentes hálózati folyamatokat ***csomagkapcsolt*** folyamatoknak is nevezik. E folyamatok esetében a csomagok a forrásállomástól a célállomásig különböző utakon juthatnak el, és lehet, hogy rossz sorrendben érkeznek meg. A forgalomirányító készülékek különböző kritériumok alapján határozzák meg a csomagok útvonalát. Ezek egy része, mint például a rendelkezésre álló sávszélesség, csomagról csomagra változhat.

Az összeköttetés-alapú hálózati folyamatokat ***vonalkapcsoltnak*** is nevezik. Ezek a folyamatok előbb kiépítenek egy összeköttetést a címzettel, és csak ezután kezdik az adatokat átvinni. Minden csomag egymás után, változatlan sorrendben, ugyanazon az áramkörön halad keresztül.

Az IP mint összeköttetés-mentes hálózati szolgáltatás.

Az IP egy összeköttetés-mentes rendszer, mely minden csomagot a többitől függetlenül kezel. Ha például egy fájlt egy FTP program segítségével töltünk le, az IP a fájlt nem egyetlen hosszú adatfolyamként küldi el, hanem csomagokra bontja, és minden csomagot külön kezel. A csomagok más-más útvonalon haladhatnak. Az is lehet, hogy egy részük elvész. Az IP a szállítási rétegre bízza a csomagvesztések megállapítását, és az esetleges újraküldés-kérést. A csomagok helyes sorbaállításáért szintén a szállítási réteg felel.

Hogyan szerezhetik be a forgalomirányítók a hálózatra vonatkozó információkat?

A forgalomirányítók kétféle módon juthatnak irányítási információkhoz: statikus, illetve dinamikus forgalomirányítással. Ha az irányítótáblába manuálisan veszünk fel bejegyzéseket, akkor statikus forgalomirányításról beszélünk. Ha viszont az útvonal-információk automatikusan kerülnek a táblába, akkor dinamikus forgalomirányításról van szó.

Forgalomirányítási algoritmus nem adaptív vagy statikus módon.

Értelmetlen dolognak tűnhet a forgalomirányító irányítótábláiba való manuális információbevitel, ha a forgalomirányító automatikusan is meg tudja "tanulni" a szükséges irányítási információkat. A kézi bevitelnek akkor lehet értelme, ha a hálózati rendszergazda befolyásolni akarja a forgalomirányító útválasztását. Például akkor lehet szükség statikus forgalomirányításra, ha egy adott vonalat akarunk tesztelni, vagy ha takarékoskodni akarunk a nagy kiterjedésű hálózat sávszélességével. Továbbá akkor is statikus forgalomirányítást célszerű használni az irányítótáblák karbantartására, ha a célhálózathoz csak egyetlen vonal vezet. Itt gyakorlatilag a véghálózatokról beszélünk, mert azok esetében a legjobb út azonos az egyetlen létező úttal.

Forgalomirányítási algoritmus adaptív vagy dinamikus módon.

Adaptív vagy dinamikus forgalomirányításról akkor beszélünk, ha a forgalomirányítók rendszeres időközönként útvonalfrissítő üzeneteket küldenek egymásnak. Minden alkalommal, amikor egy forgalomirányító egy új információkat tartalmazó üzenetet kap, az információk alapján kiszámítja a legjobb új útvonalat, majd ennek megfelelő frissítő üzenetet küld a többi forgalomirányítónak. Dinamikus forgalomirányítás használatával a forgalomirányítók képesek alkalmazkodni a hálózat változásaihoz. A dinamikus irányítótábla-frissítés megjelenése előtt a legtöbb gyártónak magának kellett az ügyfelei irányítótábláit karbantartania. Ez azt jelentette, hogy minden eladott vagy bérbe adott berendezés irányítótábláját a gyártónak kézzel kellett feltöltenie a megfelelő hálózati címekkel, a hozzá tartozó távolságértékekkel és portszámokkal. A hálózatok növekedésével ez egyre fáradságosabb és időigényesebb, vagyis egyre költségesebb feladattá vált. A dinamikus forgalomirányítás azonban megszabadítja a gyártókat és a hálózati rendszergazdákat az irányítótáblák manuális feltöltésének terhétől. A dinamikus forgalomirányítás akkor használható igazán jól, ha elegendően nagy a sávszélesség, illetve a hálózati forgalom nem túl nagy. A dinamikus forgalomirányítást megvalósító protokollokra példa a RIP, az IGRP, az EIGRP és az OSPF protokoll.

Legrövidebb út kiválasztása:

Többféle legrövidebb út lehetséges: PL Legkevesebb ugrás, Legrövidebb földrajzi távolság, Leggyorsabb átvitel.

- Általában a földrajzi távolság, költség, sávszélesség, késleltetés, aktuális forgalom együttesen határozzák meg

Legrövidebb útvonal meghatározása

- Felrajzoljuk az alhálózat gráfját
- Csomópontok: routerek
- Élek: kommunikációs csatornák (kapcsolatok)
- A gráf két csomópontja közti legrövidebb út kiszámításával

Elárasztás

- A bejövő csomagokat minden más irányba továbbítja
- Probléma: a csomagok többször érkeznek meg
- Ugrásszámláló a csomag fejrészében
- Kezdeti érték: a forrástól célig tartó ugrások száma
- Ha nem ismert, az alhálózat legnagyobb távolságára
- Elárasztással továbbküldött csomagok nyilvántartása
- Minden csomagot sorszámmal lát el a forrásrouter
- Kétszer ugyanaz a csomag ne kerüljön szétküldésre
- Ha a csomag a listában van (már elküldte) akkor eldob

15. Tétel

A távolságvektor és a kapcsolatállapot alapú forgalomirányítás működése. Az egyes megoldások összehasonlítása, az előnyök és hátrányok bemutatása. A forgalomirányítás során felmerülő problémák elemzése, a lehetséges megoldások ismertetése.

Távolság vektor alapú forgalomirányítás: Bellmann (1957), Ford és Fulkerson (1962). Az Arpanet eredeti irányító algoritmusa. Az interneten is használható, például: RIP

- **Minden** router egy táblázatban kezel az alhálózat többi routeréről, ami tartalmazza
- Az adott router távolságát: Ugrásszám, Késleltetés (speciális ECHO csomagokkal mérhető)
- Preferált kimenő vonalat az adott routerhez
- **Meghatározott** időközönként a szomszédos routerek kicserélik egymással a táblázataikat

A végtelenig számolás problémája

A távolság alapú forgalomirányítás:

Gyorsan reagál a pozitív változásokra
- vonal vagy eszköz helyreállása
- Ha a leghosszabb út N ugrás, akkor N csere múlva minden eszköz értesül a változásról

Lassan reagál az eszközök, útvonalak meghibásodásra
- vonal vagy eszköz meghibásodása

Megoldások (pl lassú konverálásra)

Látóhatár megosztás
- Egy hálózatra vonatkozó információt csak a hálózat felől fogad a router

Útvonalak mérgezése
- A kiesett útvonalhoz tartozó érték végtelenre állításával
- A szomszédos eszközök azonnal rögzítik az új értéket

Eseményvezérelt frissítések
- A topológia változásakor azonnali üzenetküldés a szomszédos eszközöknek
- A frissítési hullám a teljes hálózaton végighalad
- Az útvonalak mérgezésével együtt használható hatékonyan

Visszatartási időzítők alkalmazása
- Ha egy hálózat elérhetetlenné válásáról érkezik frissítés, elindul egy időzítő
- Ha az időzítés lejárta előtt
  - ugyanonnan helyreállásról érkezik frissítés minden rendben, helyreállt a rendszer.
  - máshonnan, de alacsonyabb értékkel érkezik frissítés az lesz az új irány
  - máshonnan, de magasabb értékkel érkezik frissítés nem veszi figyelembe
- Időt biztosít az információ elterjedésére a teljes hálózatban

A távolságvektor alapú rendszerek még az előbbi megoldásokat használva is lassan konvergálnak

Kapcsolat állapot alapú forgalom irányítás

1. A szomszédok felkutatása és felderítése
2. A szomszédok felé vezető út (késleltetés, költség) felmérése
3. Az új információkból egy csomag generálása
4. A csomag továbbítása a hálózat összes routere felé
5. A kapott csomagokból kiszámítani az utat a többi router felé

- Minden router a teljes hálózatról rendelkezik információval: Topológia, Késleltetések, költségek
- Dijkstra algoritmusa alapján meghatározzák a legrövidebb utat az összes routerhez

Irányító protokoll:

A forgalomirányítók az irányító protokollokat arra használják, hogy egymásnak átadják irányítótábláikat *(ARP tábláikat melyekben a forgalomirányítóra kapcsolt hostok IP és MAC címei vannak)*, és megosszák egymással a forgalomirányítási információkat. A forgalomirányítók egyszerre több, egymástól független irányító protokollt is képesek támogatni, és egy időben sokféle irányított protokoll irányítótábláját képesek karbantartani. Ez a tulajdonság teszi lehetővé, hogy egy forgalomirányító különböző irányított protokollok csomagjait tudja átvinni ugyanazon az összeköttetésen.

RIP protokoll

Az egy hálózaton belüli forgalomirányítók legtöbbször a *RIP protokollt* használják az irányítási információk átvitelére. Ez a célig vezető út hosszát aszerint határozza meg, hogy hány *ugrás* (azaz hány forgalomirányító) esik a csomag útjába. A RIP beállítható időközönként (általában 30 másodpercenként) frissíti a forgalomirányítók irányítótábláit. Mivel a RIP protokollt használó forgalomirányítók **szinte egyfolytában kommunikálnak** egymással, **nagy hálózati forgalmat generálnak, mely a RIP egyik hátrányaként róható fel.** A RIP révén tudják a forgalomirányítók kiválasztani, hogy melyik útvonalon küldjék az adatokat. Ehhez a *vektortávolság* fogalmát használják. Ha az adat áthalad egy forgalomirányítón, azaz egy új azonosítóval rendelkező hálózaton, azt egy ugrásnak tekintik. Például az, hogy egy útvonal *ugrásszáma* 4, azt jelenti, hogy

az ezen az útvonalon haladó csomagnak négy forgalomirányítón kell áthaladnia, amíg a célállomásig elér. **Ha a célállomáshoz több útvonalon is el lehet jutni, akkor a forgalomirányító azt az utat választja, amelyen az ugrások száma a legalacsonyabb.** Mivel a RIP a legjobb útvonal kiválasztásához az ugrásszámon kívül más irányítási mértéket nem vesz figyelembe, **nem biztos, hogy a célállomáshoz vezető leggyorsabb útvonalat választja ki.** Ennek ellenére a RIP nagyon népszerű, és ma is sok helyen használják. Ez elsősorban annak köszönhető, hogy ez volt a legelsőként kifejlesztett forgalomirányító protokoll. A RIP protokollal kapcsolatban egy másik probléma is felmerül. Ugyanis **elképzelhető, hogy egy állomás már túl messze van** ahhoz, hogy RIP protokol használatával elérhető legyen**. Ez annak a következménye, hogy a RIP maximum 15 ugrást enged meg egy csomag átviteléhez.** Ha tehát a célhálózat több mint tizenöt forgalomirányító távolságban van, akkor azt a RIP nem tekinti elérhetőnek.

IGRP

Az IGRP protokollt kifejezetten a RIP protokoll által kezelhetetlen, nagy méretű, több gyártótól származó készülékeket tartalmazó hálózatokra fejlesztették ki. A RIP-hez hasonlóan az IGRP is távolságvektor alapú protokoll, azonban az IGRP további információkat is figyelembe vesz, például a sávszélességet, a terhelést, a késleltetést és a megbízhatóságot.

EIGRP

Az EIGRP protokoll az IGRP protokoll továbbfejlesztett változata. Az EIGRP az elődjénél nagyobb hatékonysággal dolgozik, emellett ötvözi a távolságvektor és a kapcsolatállapot alapú forgalomirányítás előnyeit.

16. tétel

Torlódásvédelem a hálózati rétegben. A torlódások kezelésének elvi háttere. Torlódásvédelem virtuális áramkörök és datagram alapú hálózatok esetében. A terhelés eltávolításának módszerei. Véletlen korai detektálás. Dzsitterkezelés.

Az OSI modell **hálózati rétege**

Feladata, hogy a csomagokat a forrástól eljuttassa a célig. A feladata ennyiben nem merül ki, mivel abban az esetben, ha a célállomás több útvonalon keresztül is megközelíthető, a hálózati réteg feladata, hogy ezek közül a legoptimálisabbat kiválassza. Ez a legalsó réteg, amely a két hálózati végpont közötti átvitellel foglalkozik. A hálózati réteg a felette álló szállítási rétegnek nyújt szolgálatokat.

A szolgálatokat az alábbi szempontok figyelembevételével tervezték meg:

1. A szolgálatoknak az alhálózat kialakításától függetlennek kell lennie, de fontos, hogy a hálózati topológia ismert legyen.
2. A felsőbb rétegek, de mindenek előtt a szállítási réteg elől el kell rejteni a kommunikációban résztvevő aláhálózatok számát, típusát és a topológiáját.
3. A végpontok eléréséhez címeket használunk, amelyeknek egységes rendszert kell alkotni a felsőbb rétegek számára. Ebből a szempontból nem különíthetők el a helyi (LAN) és a nagy kiterjedésű (WAN) hálózatok sem.

Torlódásvédelem:

Hatásai: Túl sok csomag jelenléte esetén csökken a szállítási kapacitás. Az elveszett csomagok további forgalmat generálnak. Összeomolhat a rendszer.

Okai: Több bemenő vonal egy kimenő irányba. Memóriahiány. Lassú processzorok, kis sávszélességű vonalak.

A rendszer részeinek egyensúlyban kell lenni. Hajlamos önmaga gerjesztésére

A torlódásvédelem alapjai

Nyílthurkú szabályozás (vezérlés)

A probléma megelőzése gondos tervezéssel

Nem alakulhat ki torlódás, nincs szükség futás közbeni beavatkozásra

Típusai

- Forrásnál beavatkozó algoritmusok
- Célnál beavatkozó algoritmusok

Zárthurkú szabályozás (visszacsatolás)

A rendszer folyamatos figyelése, a torlódás észlelése

Az információ továbbítása a beavatkozás helyére, beavatkozás

Típusai

- Explicit visszajelzéssel működő algoritmusok
- Implicit visszajelzéssel működő algoritmusok

Torlódásvédelem virtuális áramkörök (VÁ) esetén:

Belépés ellenőrzés: Ha ismert a torlódás ténye, nem épülhet fel több VÁ. Ez azonban sikertelen kapcsolat felépítést eredményez a szállítási rétegben. Drasztikus de egyszerű és hatékony megoldás. Ha a torlódás megszűnt, felépíthetők az új VÁ-ök.

A torlódott csomópontok elkerülése: A VÁ felépítése csak a torlódást elkerülve, alternatív módon történhet meg. Ha nincs más útvonal, akkor nem épülhet fel új VÁ.

Erőforrások foglalása a VÁ felépítésekor: Ha a szükséges erőforrások garantáltak, nem léphet fel torlódás.

Torlódásvédelem datagram alapú hálózatokban:

Kimenő vonalak kihasználtságának figyelése: küszöbérték felett beavatkozás.

Figyelmeztető bit használata: A kimenő csomag fejrészében egy bit beállítása torlódáskor. A nyugtába a cél bemásolja a figyelmeztető bit értékét. A forrás közben folyamatosan csökkenti a sebességet, amíg a bit igaz értékű marad. A sebesség újra növelhető ha a bit hamisra vált. Ezzel a probléma: az átvitel során bármely router elhelyezhette a bitet, a forgalom csak akkor nőhet ha sehol nincs torlódás.

Lefojtó csomagok: A torlódást érzékelő router azonnal jelez az adónak, lefojtó csomagot küld a célcsomópont megjelölésével. A továbbított csomagot is megjelölik a fejrészben. A forrás csökkenti a sebességet a lefojtó csomag hatására.

Lefojtás lépésről lépésre: Nagy távolság esetén hosszú reakcióidő, a lefojtás az előző routerre hat, lépésenként ér a forráshoz.

A terhelés eltávolítása:

Fő kérdés: melyik csomagot dobjuk el? **3** megoldás:

Csomagok eldobása véletlenszerűen.

Csomagok eldobása sorszám alapján: újabb csomagok eldobása („bor politika”: a régebbi csomag értékesebb az újnál). Régebbi csomagok eldobása: („tej politika”: Az új csomag jobb mint a régi)

Intelligens csomageldobás: Az alkalmazásnak prioritással kell ellátnia a csomagot, először az alacsonyabb prioritású csomagokat dobja el.

Véletlen korai detektálás:
A torlódás kialakulását célszerű megelőzni egy küszöbérték átlépésekor, amíg kezelhetőbb a helyzet.
Bizonyos szállítási protokollok az adás lassításával reagálnak az elveszett csomagokra (pl. TCP).
Kérdés, hogy melyik forrás okozza a feltorlódott kimenő sort? Általában nem egyszerű kideríteni, a sorból véletlenszerűen dobálunk el csomagokat. Az érintett források csökkentik az adási sebességet. Nem alkalmazható vezetéknélküli hálózatokban, ott jellemzően a rádiós probléma okozza a csomagvesztést.
Dzsitterkezelés:
A késleltetés nagyságánál sokkal lényegesebb a késleltetés ingadozása, pl. webes rádiók, tv-knél. Lényegtelen hogy 25ms vagy 55ms késleltetéssel érkezik az adás, csak az a fontos, hogy azonos időközönként érkezzenek a képkockák. Megadható a maximális és minimális késleltetések értéke (dzsitter).
A késleltetés befolyásolása:
- Pufferelés a vevő oldalán: ezt valós idejű interaktivitást igénylő alkalmazások nem engedik meg.
- Minden ugrásra kiszámoljuk az átviteli időt a teljes út mentén. Ha a csomag siet, az adott router hosszabb ideig puffereli. Ha késik, akkor igyekszik a lehető leghamarabb továbbítani.

17. tétel

A QoS jellemző paraméterei, különböző felhasználási területek igényei. A QoS biztosításának lehetséges eszközei: túlméretezés, pufferelés, forgalomformálás, erőforrás lefoglalás, belépés engedélyezés, arányos útvonalválasztás, csomagütemezés. A lyukas vödör és a vezérjeles vödör algoritmusok.

Az OSI modell **hálózati rétege**

Feladata, hogy a csomagokat a forrástól eljuttassa a célig. A feladata ennyiben nem merül ki, mivel abban az esetben, ha a célállomás több útvonalon keresztül is megközelíthető, a hálózati réteg feladata, hogy ezek közül a legoptimálisabbat kiválassza. Ez a legalsó réteg, amely a két hálózati végpont közötti átvitellel foglalkozik. A hálózati réteg a felette álló szállítási rétegnek nyújt szolgálatokat.

A szolgálatokat az alábbi szempontok figyelembevételével tervezték meg:

1. A szolgálatoknak az alhálózat kialakításától függetlennek kell lennie, de fontos, hogy a hálózati topológia ismert legyen.
2. A felsőbb rétegek, de mindenek előtt a szállítási réteg elől el kell rejteni a kommunikációban résztvevő aláhálózatok számát, típusát és a topológiáját.
3. A végpontok eléréséhez címeket használunk, amelyeknek egységes rendszert kell alkotni a felsőbb rétegek számára. Ebből a szempontból nem különíthetők el a helyi (LAN) és a nagy kiterjedésű (WAN) hálózatok sem.

QoS:

Hálózatok és hálózati eszközök képessége az erőforrások meghatározott rend szerinti felosztására, és garantált sávszélesség biztosására. A QoS-t támogató hálózatokon a magas prioritású üzenetek előnyben részesíthetők alacsonyabb besorolású társaikkal szemben, és konkurrencia-helyzetben előbbiek továbbítása utóbbiak feltartóztatásával garantált sebességen biztosítható.

Paraméterei:

legfontosabb jellemzők: Megbízhatóság, késletetés, dzsitter, sávszélesség.

Felhasználási területei: eltérő igények a különböző alkalmazásoknál, különböző átviteli utak és eszközök.

A QoS biztosításának eszközei:

Túlméretezés: Pontosan kell ismerni az igényeket, de ezek idővel változhatnak. Megjelenhetnek átlagostól eltérő csúcsértékek is. Általában drága.

Pufferelés: növeli a késleltetést, de csökkenti a dzsittert (Megadható a maximális és minimális késleltetések értéke).

Forgalomformálás: Az adás átlagos sebességének szabályozása, kapcsolat felépítésekor szolgáltatásszíntű megállapodás, forgalmi rendfenntartás.

Erőforrás lefoglalása: Minden csomag számára szükséges egy kijelölt út, a kijelölt út mentén lefoglalhatók az erőforrások (sávszélesség, processzoridő, puffer).

Belépés engedélyezés: Kérdés: biztosíthatók-e egy adott folyam igényei? A forrás és a cél között minden eszköznek vizsgálnia kell, a folyamnak pontosan kell leírni az igényeit.

Folyammeghatározás: fontosabb paraméterei: vezérjeles vödör sebessége, mérete. Adatsebesség csúcsértéke, minimális és maximális csomagméret.

Arányos útvonalválasztás: Az azonos célba tartó csomagok szétosztása alternatív utakon (egyenlő részekre osztás, vagy felosztás a kimenő vonalak kapacitásának függvényében).

Csomagütemezés: Ha egy kimenő sort használ, akkor egyetlen folyam nagy sávszélességre tehet szert, más folyamok szolgáltatásminősége nehezen biztosítható. Egyenlő esélyű sorbaállítás esetén minden folyam saját várakozó sort használ, körforgás a várakozó sorok között. Hátránya, hogy nagyobb csomagméret esetén nagyobb sávszélességet igényel, erre megoldás a bájtonkénti körforgás szimulálása. Súlyozott egyenlő esélyű sorbaállítás esetén prioritás biztosítható bizonyos folyamoknak.

A lyukas vödör és a vezérjeles vödör algoritmusok:

Lyukas vödör algoritmus: Minden hoszt egy véges sort tartalmazó interfésszel kapcsolódik. Csomag érkezése: ha a sorban van hely, a csomag a sor végére kerül. Ha nincs hely, a csomag eldobásra kerül. A hoszt minden órajelkor azonos számú csomagot küld tovább. Ha azonos a csomagméret, akkor órajelenként egy csomagot, ha változó, akkor bájtszámlálás.

Vezérjeles vödör algoritmus: Szükség esetén változtatható sebesség a kimeneten: rövid ideig megengedi a kimenet gyorsítást, célja az adatvesztés elkerülése lökésszerű terhelés esetén. Nem dob el csomagot, ha megtelik a vödör, akkor a hosztot utasítja a küldés felfüggesztésére.

18. tétel

Valós átviteli utak, összekapcsolt hálózatokon. Hálózatok összekapcsolásának problémái eszközei. Virtuális áramkörök és datagram alapú hálózatok összekapcsolása. Alagutak használata. Csomagok tördelése, a tördelés problémái. Darabok számozása.

Az OSI modell **hálózati rétege**

Feladata, hogy a csomagokat a forrástól eljuttassa a célig. A feladata ennyiben nem merül ki, mivel abban az esetben, ha a célállomás több útvonalon keresztül is megközelíthető, a hálózati réteg feladata, hogy ezek közül a legoptimálisabbat kiválassza. Ez a legalsó réteg, amely a két hálózati végpont közötti átvitellel foglalkozik. A hálózati réteg a felette álló szállítási rétegnek nyújt szolgálatokat.

A szolgálatokat az alábbi szempontok figyelembevételével tervezték meg:

1. A szolgálatoknak az alhálózat kialakításától függetlennek kell lennie, de fontos, hogy a hálózati topológia ismert legyen.
2. A felsőbb rétegek, de mindenek előtt a szállítási réteg elől el kell rejteni a kommunikációban résztvevő aláhálózatok számát, típusát és a topológiáját.
3. A végpontok eléréséhez címeket használunk, amelyeknek egységes rendszert kell alkotni a felsőbb rétegek számára. Ebből a szempontból nem különíthetők el a helyi (LAN) és a nagy kiterjedésű (WAN) hálózatok sem.

Eltérő hálózatok használata:

Különböző hálózatok használatának okai: eltérő igények, technológiák. Különböző hardverek és szoftverek. Esetleg alacsonyabb telepítési költségek.

Különbségek lehetnek: szolgálat típusa, minősége, hibakezelés, címzés, protokollok, csomagméretek, forgalomszabályozás, torlódásvédelem.

Hálózatok összekapcsolásának eszközei:

A **fizikai** rétegben: HUB vagy reapeater azonos típusú hálózatok között továbbítják a biteket.

Az **adatkapcsolati** rétegben: Bridge vagy Switch keretek továbbítása különböző hálózatokba (MAC). Egyszerű protokoll konverziók pl Ethernet ->802.11.

A **hálózati** rétegben: Router több protokollt kezel, esetenként a csomagformátumokat is átalakíthatja.

A **szállítási** rétegben: Szállítási átjárók gondoskodnak két eltérő szállítási protokollal rendelkező hálózat összekapcsolásáról.

Az **alkalmazási** rétegben: Alkalmazási átjárók.

Virtuális áramkörök összekapcsolása:

Az összekapcsolt hálózat virtuális hálózat virtuális áramkörök sorozata. Távoli cím esetén az alhálózat VÁ-t épít ki az első routerig, majd az adott router a következő routerig, egészen az utolsó router és a célhálózat közötti VÁ-ig. A routerek között lehetnek többprotokollos routerek is (átjárók). A routerek feladata: táblázatokban jegyzik a rajtuk áthaladó virtuális áramköröket, az egyes VÁ-ök továbbításának irányát, az új VÁ számát.

Datagram alapú hálózatok összekapcsolása:

Az egyes alhálózatokat routerek kapcsolják össze, a csomagok különböző útvonalakon haladnak, ezzel kivédik a meghibásodásokat. Forgalom és terhelés figyelése. Eltérő protokollok a hálózati rétegben: konverzió a többprotokollos routerek segítségével, útvonalak keresése azonos típusú alhálózatokat használva. *Címzés*: Problémás a nagyméretű, mindenre kiterjedő leképzési táblák használata, ezért univerzális csomagokat (IP) használ.

Alagutak használata:

Azonos típusú hálózatok összekötése más típusú hálózat(ok) felhasználásával.

Példa: A forrás és a cél TCP/IP-t használó Ethernet hálózat hosztja. A forrás a cél IP címével Ethernet keretet küld. Az első router kiveszi az IP csomagot a keretből és a megfelelő módon és formában a cél hálózat routerének címzi. A másik router kiveszi a kapott csomagból az IP csomagot és egy Ethernet keretbe ágyazva továbbítja a célhosztnak. A routertől-routerig tartó szakasz egy soros vonalként fogható fel. Ezen a vonalon az IP csomag beágyazva halad a WAN csomaghoz tartozó adatmezőben. A WAN-on történő átvitelben az eredeti csomagnak nincs szerepe.

Csomagok tördelése:

Különböző hálózatokban különböző a csomagok mérete, pl ATM:48 bájt, IP (maximum) 65515bájt. Nem engedjük a nagyobb csomagokat a kisebb csomagméretet használó hálózatok irányába (ez csak elvben működik, de a gyakorlatban nem megoldás). A nagyobb csomagokat darabokra kell tördelni, majd azokat külön csomagként továbbítani. A darabokat később újra össze kell állítani.

Kétféle darabolás:

Transzparens darabolás: a kimenő átjárónak tudnia kell, mikor kapott meg minden darabot. Minden csomagnak ugyan azon az átjárón kell végződnie. Probléma: a darabolás és összeállítás többszöri elvégzése késleltetést okoz.

Nem transzparens darabolás: Amit egyszer feldaraboltunk, azt utána önálló csomagként kezeljük. De a darabokhoz kapcsolódó kiegészítő információk miatt megnő az adatmennyiség. A darabok összerakása a célnál történik. Minden hosztnak képesnek kell lennie a darabok összeállítására. Több átjáró, különböző útvonalak használhatók.

Darabok számozása:

Fa struktúra: első darabolásnál 0.0, 0.1, 0.2, stb. Ha további darabolás szükséges akkor: 0.1.0, 0.1.1, 0.1.2 stb.

Elemi darabméret meghatározásával: Elégségesen kis méret minden hálózaton való áthaladásához. Minden darab mérete megegyezik, kivéve az utolsót. Az összeállításhoz minden darab tartalmazza a csomag számát, a csomagban lévő első elemi darab számát, és a csomag végét jelző bitet.

19. tétel

Az IP bemutatása az IP fejrész mezői alapján. Az egyes mezők szerepének, felépítésének, jellemzőinek bemutatása. (A fejrész ábráját az oktató biztosítja)

Az IP fejrész

IP csomag

- Fejrész
- Adatrész

Verzió (4 bit)

- Az adott csomag a protokoll melyik verziójához tartozik
  - Folyamatos áttérés IPv4-r_l IPv6-ra
  - Értéke jellemz_en 4 (IPv4) vagy 6 (IPv6)

IHL (4 bit)

- A fejrész hossza 32 bites szavakban
- Megadja az opció maximumát (40 bájt)

Szolgálat típusa (6 bit)

Régen

- 3 bit - precedencia mező
  - Értéke: 0 (normál csomag), 1 (prioritásos) … 7 (vezérlőcsomag)
- 3 jelzőbit: D (delay), T (Throughput), R (Reliability)
- 1 jelzőbit: C (cost)
- 1 bit: MBZ

Napjainkban

- DSCP (Differentiated Services Code Point) mez_ (6 bit)
- IP-prioritást és a szolgáltatástípust jelölő mezők kombinációja
- A QoS biztosításban játszik szerepet a mező

Teljes hossz (16 bit)

- A fejrész és az adatrész együttes hossza
- Egy datagram maximális mérete: 65.535 bájt

Azonosítás (16 bit)

- A datagram darabjainak azonosítására
- A datagram minden darabja ugyan azt az azonosítót kapja

DF: Don't Fragment (1 bit)

- Jelzi a routerek számára, hogy ne darabolják a datagramot
- Szükséges lehet néhány (kiscsomagos) hálózat elkerülése

MF: More Fragment (1 bit)

- Minden darabnál az értéke 1 kivéve az utolsó darabot (0)

Darabeltolás (13 bit)

- A darab helyének meghatározása a datagramban
- Elemi darabméret: 8 bájt
- Meghatározza az IP csomag maximális méretét (65.536 bájt)

Élettartam (8 bit)

- Az ugrások számlálására
  - Értéke minden ugrásnál 1-el csökken
  - 0-nál eldobja a router a csomagot
- Az egyes csomagok nem keringhetnek a végtelenségig

Protokoll (8 bit)

- A feldolgozó szállítási folyamat azonosítására

Fejrész ellenőrző összeg (16 bit)

- A fejrész mezőiben történt hibák jelzésére
- Minden ugrásnál újra kell számolni!

Forrás címe (32 bit)

Cél címe (32 bit)

Opciók

- Olyan információk melyekre csak ritkán van szükség
  - Nem foglalnak (feleslegesen) állandó helyet a fejrészben
  - Fejlesztési, kísérleti folyamatokban jól használható
  - Az eredeti fejrészből "kifelejtett" paraméterek pótlása

Az IP fejrész opciói

Biztonság

_ Az információ titkosságáról szolgál információval
_ A routerek jellemzően nem foglalkoznak az értékével
_ "Hasznos" lehet a figyelem felkeltésére

Szigorú forrás általi forgalomirányítás

_ A teljes utat adja meg IP címek sorozatával a forrástól a célig
_ A csomag útja pontosan meghatározható
_ Az irányítótáblák összeomlása esetén is küldhet_k csomagok
_ Lemérhet_ egy útvonal késleltetése

Laza forrás általi forgalomirányítás

_ A felsorolt routereken a felsorolás sorrendjében kell áthaladni
_ Más routerek is érinthet_k
_ Egyes helyek, útvonalak érinthet_k vagy kikerülhet_k

Útvonal feljegyzése

_ Minden router az opció végére fűzi az IP címét
_ A csomag útja pontosan követhető, elemezhető
_ Megtalálhatók a hibás forgalomirányítási döntések
_ Problémát jelenthet a fejrész opció mezőjének korlátozott mérete

Időbélyeg

_ Az útvonal feljegyzése opcióval azonosan működik
_ A router IP címe mellett az időbélyeget is feljegyzi

20. tétel

IP címek és címosztályok. Az IP címek felépítése és szerepe. Fenntartott címtartományok, különleges IP címek. Az IPv4 problémái és lehetséges megoldások. Alhálózatok kialakítása, az alhálózati maszk. CIDR és NAT

Az IP-cím

Egyedi kombináció, nincs két azonos IP cím. A hálózat és a hoszt számát kódolja. 32 bit hosszú címek. Megtalálhatók az IP csomagok forrás és cél mezőiben. Egy hálózati interfészre utal (egy hosztnak több IP címe is lehet). A hálózatazonosító minden IP-címben azt a hálózatot azonosítja, amelyre a készülék csatlakozik. Az IP-cím állomáscím része pedig a hálózatra csatlakozó eszközt azonosítja. Mivel az IP-címek négy, pontokkal elválasztott oktettből(8bit) állnak, ezekből egy, kettő vagy három használható hálózatazonosítóként. Hasonló módon egy, kettő vagy három oktett használható állomáscímként az IP-címben.

Az IP-címosztályok

Az InterNIC-től kapott IP-címek három osztályba sorolhatók: A, B és C osztályba. Az InterNIC az A osztályú címeket a világ kormányzatai számára, a B osztályú címeket a közepes nagyságú vállalatok számára foglalja le, mindenki más pedig C osztályú címeket kap. Az A osztályú címek bináris formájának első bitje mindig 0. A B osztályú címek első két bitje mindig 10, a C osztályú címek első 3 bitje pedig mindig 110. A osztályú IP-cím például a 124.95.44.15 cím. Az első oktett (124) az InterNIC által megszabott hálózatazonosítót mutatja. A további 24 bitet a hálózat belső rendszergazdái választják meg, illetve osztják ki. Egyszerűen eldönthető, hogy egy készülék A osztályú hálózathoz tartozik-e, ha megvizsgáljuk IP-címének első oktettjét. Az A osztályú címek első oktettje ugyanis 0 és 127 közötti értékű. Az A osztályú IP-címeknél csak az első 8 bitet használják a hálózat azonosítására. Az IP-cím további három oktettjét az állomáscím részére foglalják le. A legkisebb lehetséges állomáscímet úgy kapjuk meg, ha mindhárom oktett mind a 8 bitje 0. A legnagyobb lehetséges állomáscímet pedig úgy kapjuk meg, ha mindhárom oktett mind a 8 bitje 1. Egy A osztályú IP-címmel rendelkező hálózatban legfeljebb 2 a 24-diken (2^{24}), pontosabban 16 777 214 lehetséges IP-cím osztható ki a hozzá kapcsolódó készülékek között. A B osztályú IP-címek egy példája a 151.10.13.28 cím. Az első két oktett az InterNIC által megszabott hálózatcímet mutatja. A további 16 bitet a hálózat belső rendszergazdái választják meg, illetve osztják ki. Egyszerűen eldönthető, hogy egy készülék B osztályú hálózathoz tartozik-e, ha megvizsgáljuk IP-címének első két oktettjét. A B osztályú IP-címek első oktettjének értéke mindig 128 és 191 közé, második oktettje pedig mindig 0 és 255 közé esik. A B osztályú IP-címeknél az első 16 bitet használják a hálózat azonosítására. Az IP-cím maradék két oktettjét az állomáscím részére foglalják le. Egy B osztályú IP-címmel rendelkező hálózatban legfeljebb 2 a 16-dikon (2^{16}), pontosabban 65 534 lehetséges IP-cím osztható ki a hozzá kapcsolódó készülékek között. A C osztályú IP-címek egy példája a 201.110.213.28 cím. Az első három oktett az InterNIC által megszabott hálózatazonosítót mutatja. A további 8 bitet a hálózat belső rendszergazdái választják meg, illetve osztják ki. Egyszerűen eldönthető, hogy egy készülék C osztályú hálózathoz tartozik-e, ha megvizsgáljuk IP-címének első három oktettjét. A C osztályú IP-címek első oktettjének értéke mindig 192 és 223 közé, második és a harmadik oktettjének értéke pedig 1 és 255 közé esik. A C osztályú IP-címeknél az első 24 bitet használják a hálózat azonosítására. Csak az IP-cím utolsó oktettje szolgál az állomáscím tárolására. Egy C osztályú IP-címmel rendelkező hálózatban legfeljebb 2 a 8-dikon (2^{8}), pontosabban 254 lehetséges IP-cím osztható ki a hozzá kapcsolódó készülékek között.

Fenntartott címtartományok

Routerek nem engedik az Internet felé, otthoni hálózathoz is használhatók

- Loopback 127.x.x.x
- A osztály esetén: 10.x.x.x
- B osztály esetén: 172.16.x.x - 172.31.x.x / 169.254.x.x
- C osztály esetén: 192.168.x.x

A hálózatazonosító

Fontos, hogy megértsük az IP-cím hálózati részének, a *hálózatazonosítónak* a jelentőségét. Az állomások, illetve készülékek csak az azonos *hálózatazonosítóval* rendelkező készülékekkel tudnak kommunikálni. Még ha fizikailag ugyanazon a szegmensen is vannak, de a hálózatazonosítójuk különbözik, nem tudnak egymással kommunikálni, hacsak nincs egy további eszköz, amely képes a különböző *hálózatazonosítókat* vagy logikai szegmenseket összekötni.

A szórási cím

Az *üzenetszórási cím* olyan cím, melynek állomáscím részében csupa 1-esek állnak. Ha egy *üzenetszórási címre* küldött csomag kerül a hálózatra, azt minden állomás észreveszi. Például a 176.10.0.0 azonosítójú hálózat *üzenetszórási címe*, mellyel minden állomáshoz eljut a csomag, 176.10.255.255.

Az *üzenetszórási cím* a nagybani levélküldéshez hasonlít. Itt a levél az irányítószám alapján jut el a megfelelő területre, majd az egyes címek további része alapján jut el a címzetthez. Az *üzenetszórási cím* ugyanígy működik. A hálózatazonosító azonosítja a szegmenst, a cím további része pedig azt mondja meg a készüléknek, hogy ez egy *üzenetszórásos üzenet*, melyre minden eszköznek figyelni kell. Minden hálózati készülék felismeri a saját IP-címét,

valamint az adott hálózat *üzenetszórási címét*.

Az alhálózat fogalma

A hálózati rendszergazdáknak a nagyobb rugalmasság érdekében néha több részre kell osztaniuk a hálózatot. Különösen a nagy hálózatokat kell kisebb hálózatokra, ún. *alhálózatokra* bontani. Elsősorban azért használunk alhálózatokat, hogy csökkentsük az üzenetszórási tartományok méretét. A szórt üzeneteket ugyanis a hálózat vagy alhálózat minden állomása veszi. Ha a hálózati rendszergazda úgy véli, hogy az üzenetszórásos forgalom a sávszélességnek túl nagy részét foglalja le, csökkentheti az adott üzenetszórási tartomány méretét. Az alhálózati cím tartalmazza a hálózat azonosítóját, az alhálózat hálózaton belüli azonosítóját és az állomás alhálózaton belüli azonosítóját. A címzés harmadik (közbülső) szintje további rugalmasságot biztosít a hálózati rendszergazdák számára. Az alhálózati cím létrehozásához a hálózati rendszergazda az állomásazonosító mezőből vesz el néhány bitet, és az alhálózat mezőhöz rendeli őket. Az alhálózati címhez legalább 2 bitet kell felhasználni. Ha ugyanis csak 1 bitet vennénk el az alhálózat létrehozására, akkor csak a hálózatcím - a .0 hálózat - és az üzenetszórási cím - a .1 hálózat - állna rendelkezésünkre. Legfeljebb annyi bitet vehetünk el, hogy legalább 2 bit maradjon az állomásazonosító számára. Az *alhálózati maszk* vagy *maszk* (hagyományos nevén *kiterjesztett hálózati előtag*) azt mutatja meg a hálózati készülékek számára, hogy a cím melyik része a hálózati előtag, melyik része az alhálózati cím és melyik az állomásazonosító. Az *alhálózati maszk* 32 bit hosszú.

Az alhálózat címének meghatározásához a forgalomirányító logikai ÉS műveletet hajt végre az IP-cím és az alhálózati maszk között. Az eredmény a hálózati/alhálózati azonosító.

Az alhálózatok létrehozásához felhasználható bitek száma

Az alhálózati maszkok az IP-címekkel egyező formátumúak. 32 bit hosszúak, és 4 oktettre osztjuk őket. Az alhálózati maszk hálózati és alhálózati része csupa 1-esekből áll, míg az állomásazonosító része csupa 0-át tartalmaz. Alapértelmezésben, amikor egyetlen bitet sem vettünk el, egy B osztályú hálózat alhálózati maszkja: 255.255.0.0. Ha azonban 8 bitet elveszünk, a B osztályú hálózat alhálózati maszkja 255.255.255.0 lesz. Mivel a B osztályú hálózatokban az állomásazonosító csak kétoktettes, legfeljebb 14 bit vehető el az alhálózat számára. A C osztályú hálózatokban az állomás mező csak egyoktettes, ezért legfeljebb 6 bit vehető el alhálózat létrehozása céljából. Bármelyik osztályról van is szó, legalább 2 bitet kell elvenni. Ugyanis az olyan alhálózat, amely csak egy hálózati és egy üzenetszórási címet tartalmaz, használhatatlan. Ezért, ha csak 1 bitet vennénk el az állomásazonosítóból, két egyformán használhatatlan alhálózatot hoznánk létre. A maszk egy oktettjének értéke attól függ, hogy hány bitet használunk fel az adott bájtból. Minden oktettből a legnagyobb helyértékű bitet vesszük. E bitek decimális értékét használjuk az alhálózati maszk kiszámításához. Ha például 1 bitet használunk, az oktett értéke 128 lesz. Ha 2 bitet használunk, az oktett értéke 192 lesz (128+64).

Az alhálózatokra kapcsolható állomások számának meghatározása az alhálózati maszk és az IP-cím alapján

Ha biteket veszünk el az állomás mezőből, fontos tudnunk, hány hálózatot hozunk létre adott számú bit elvételével. Már említettük, hogy 1 bit nem vehető el, legalább 2 bitet kell elvennünk. 2 bit elvételével 4 (2^2) hálózatot hozunk létre. Egy újabb bit elvételével a létrejövő alhálózatok száma a kétszeresére növekszik. 3 bit elvételével 8, azaz 2^3 alhálózat keletkezik. 4 bit elvételével pedig 16, azaz 2^4 alhálózat jön létre. A jobb érthetőség kedvéért vegyünk egy C osztályú hálózatazonosítót! Ha nem használunk alhálózati maszkot, az utolsó oktettnak mind a 8 bitje az állomások azonosítására szolgál. Ezért 256 (2^8) lehetséges cím osztható ki az állomások között. Most képzeljük el, hogy ezt a C osztályú hálózatot két alhálózatra bontjuk! Ha 1 bitet vennénk el az állomás mezőből, 7 használható bit maradna. Ha a megmaradó 7 bit minden lehetséges bitkombinációját felírnánk, azt látnánk, hogy alhálózatonként összesen már csak 128 (2^7) állomás címezhető meg. Ha ugyanezen hálózatban 2 bitet vennénk el az állomás mezőből, akkor már csak 6 bit maradna az állomások azonosítására. Ekkor 64-re csökkenne (2^6) a kioszthatóállomáscímek száma. Az egy alhálózaton kioszthatóállomáscímek száma attól függ, hogy hány alhálózatot hoztunk létre. Például ha egy C osztályú cím esetén a 255.255.255.224-es alhálózati maszkot használjuk, akkor 3 bitet vettünk el az állomás mezőből, és 8, egyenként 32 állomáscímmel rendelkező alhálózatot hoztunk létre.

IPv4 címprobléma

- Az IPv4 nem jelzi a földrajzi távolságokat, ami pedig nagyon hasznos lenne az útvonalválasztás optimalizálásánál.
- A nagyméretű site-k több C osztályú blokkot igényelnek, ami miatt az interdomain routing táblák gyorsabban nőnek, mint a router memóriája
- A felosztott címtér kezelése drága és összetett feladat.
- 3 Millió Web Site (. Jan 1999)
- 700+ Millió weboldal
- 8000 ISP világ szinten (4700+ U. S.);
- A forgalom növekedése 100- 1000%/ évente
- 300 M - 1000 M felhasználó 2002-re

Saját címek - NAT

Minden IP-címtartományban van néhány olyan cím, amelyet az InterNic nem oszt ki. Ezeket a címeket *saját címeknek* hívják. Akkor is saját címeket szokás kiosztani, ha nincs elég nyilvános cím. A saját rendszereknek a nyilvános hálózatokhoz való csatlakoztatásához *hálózati címfordító (NAT)* kiszolgáló vagy *proxy* kiszolgáló használható, valamint néhány nyilvános cím is szükséges. **Network Address Translation** (NAT, magyarul: hálózati címfordítás): A hálózati címfordító a belső gépekről érkező csomagokat az Internetre továbbítás előtt úgy módosítja, hogy azok feladójaként saját magát tünteti fel, így az azokra érkező válaszcsomagok is hozzá kerülnek továbbításra, amiket – a célállomás címének módosítása után – a belső hálózaton elhelyezkedő eredeti feladó részére továbbít.

A CIDR lényege, hogy szakít az A, B és C címosztályok koncepciójával. Helyette a hálózati prefix, hálózati maszk koncepcióját általánosítja. Az Internet routerek nem az IP cím első három bitje alapján állapítják meg, hogy hol húzódik a határ a hálózati cím és az állomáscím között, hanem e helyett a routing protokollok magukkal hordoznak egy bitmaszkot, amely a hálózati előtag hosszát adja meg. A CIDR-t ismerő routing protokollok nem törődnek a címosztállyal, mindössze a maszkot figyelik. Elemzők szerint, ha 1994/95-ben nem vezetik be a CIDR technológián alapuló címkiosztást, a routing táblák akkorára nőttek volna, hogy az Internet mára működésképtelen lenne. A legtöbb router ma már ismeri ezt a technikát, és jelenleg az IANA is CIDR alapján osztja ki a címeket

21. tétel

Az IPv6 működésének, előnyeinek az elterjedés lehetőségeinek bemutatása. Az IPv6 fejrész elemzése, az egyes mezők szerepének, felépítésének, jellemzőinek bemutatása. (A fejrész ábráját az oktató biztosítja)

- Mi az IPV6?

Az IPv6 vagy másnéven Internet Protokoll 6 az internet új generációs protokollja. Az IETF (Internet Engineering Task Force) - egy testület, amely felügyeli és ellenőrzi az Internetet - kifejlesztette az IPv6-ot ami a már 20 éves IPv4-et fogja felváltani. Az IPv4 egy ideig rugalmasnak bizonyult, azonban mára már tapasztalhatjuk a korlátait. Különösen kritikus probléma az, hogy az új internetes eszközök (a 3G technológia és a "mobil" internet) következtében "kifogyunk" az internet címekből. A jövő ugyanis az hogy minden technikai eszközt irányítani tudunk majd valamilyen kezelőfelületről, és ehhez az kell hogy minden eszköznek legyen egy címe.

- Mi a különbség az IPv4 es az IPv6 között?

Az elsődleges differencia az hogy az IPv6 128 bit-es címet használ a megszokott 32 bites IPv4-es címek helyett. Ez 2^96 - (durván 7,92 x 10^28) szorosára növeli az elméletben rendelkezésre álló címtartományt.

Elméletileg a Föld minden egyes homokszemcséjének jutna egy IPv6-os cím. Lényeges, hogy megszüntek különböző méretű hálózatokon alapuló tartományok (A, B és C tartományok). Az IPv6 továbbá olyan lehetőségeket is nyújt, mint például a hitelesítés és biztonság, Quality-Of-Service, Anycast címzési mód. Alapjában az IPv6 megőrzi az IPv4 és protokoll alaptulajdonságait. Továbbra sem változott az alapelv, hogy az IP-címeket nem gépekhez, hanem halózati interfacekhez rendelik. Egy interfacenek viszont több címe is lehet, és ezt az új szolgáltatások ki is használják. Az új címeket a hozzárendelés elve szerint három alapvető csoportba oszthatjuk: egycélú (unicast), választható célú (unicast) és a többcélú (multicast) címek.

Az IPv6 előnyei:

Megtartotta az IPv4 előnyeit, kompatibilis a legtöbb ipv4-es protokollal. Az ipv4-gyel nem kompatibilis, de párhuzamosan használhatók.

Az IPv6 legfontosabb fejlesztései: „kifogyhatatlan”, hosszabb (128bites) címek. Az egyszerűbb fejrésznek köszönhetően gyorsabb feldolgozhatóság a routerekben. Az opciók továbbfejlesztett támogatása: néhány eddig állandó érték opcióként jelenik meg, az opció fölösleges részei könnyen átléphetők (gyors feldolgozás).

A biztonság fokozása: hitelesítés és titkosság biztosítása. A szolgálat típusának és a szolgálatminőségnek kiemelt kezelése.

Az IPv6 csomag fejrésze:

Verzió	Prioritás	Folyamcímke	
Adatmező hossza		Következő fejrész	Ugráskorlát
Forrás címe			
Cél címe			

Az IPv6 fejrész:

Verzió: 4bit. Tartalma: Az adott csomag a protokoll melyik verziójához tartozik. Értéke jellemzően 4 (IPv4) vagy 6 (IPv6). (folyamatos áttérés ipv4-ről ipv6-ra).

Forgalmi osztály: 8bit. Az adott forgalom prioritási osztályba sorolásához. A valós idejű szállítási követelmények besorolása.

Folyamcímke: 20bit. Egy folyamathoz tartozó csomagok útjának azonosítása a forrástól a célig. Az egyes címkékhez különböző igények tartozhatnak a routerekben: erőforrások foglalása, késleltetéssel kapcsolatos igények kezelése.

Adatmező hossza: 16bit. Az adatmező mérete bájtokban (fejrész nélkül).

Következő fejrész: 8bit. A fejrész egyszerűsítését biztosító mező, további kiegészítő fejrészeket azonosíthat. Ha nincs további fejrész, akkor a szállítási protokollt azonosítja.

Ugráskorlát: 8bit. Az IPv4 élettartam mezőjével azonos funkció. Értéke minden ugrásnál csökken (0-nál eldobja a csomagot).

Forrás és cél címe: 128bit:

Új formátum: nyolc csoportban négy-négy hexadecimális számjegy, elválasztó karakter „ : ” Pl: 1080:0000:0000:0000:0008:0800:200C:417A.

Egyszerűsítések: Vezető nullák minden csoport elején elhagyhatók. A csak nullát tartalmazó csoportok kettősponttal helyettesíthetők. A „ :: „ kombináció minden címben csak egyszer szerepelhet. Pl.: 1080::8:800:200C:417A. IPv4-es címek írásmódja::193.6.50.195

22. tétel

Az UDP felépítése és jellemzői. A TCP feladatai és jellemzői, a TCP portok szerepe. A TCP protokoll legfontosabb feladatai, sorszámozás, nyugtázás. A TCP fejrész elemzése az egyes mezők szerepének, jellemzőinek alapján. (A fejrész ábráját az oktató biztosítja)

Szállítási réteg

A szállítási (4.) réteg feladatának meghatározásakor gyakran emlegetik a "szolgálat minősége" (QoS: Quality of Service) kifejezést. A réteg elsődleges feladata az adatoknak a forrásállomástól a célállomásig való megbízható és pontos eljuttatása, valamint az adatfolyam szabályozása. A végpontok közötti forgalomszabályozást a szállítási réteg az ún. csúszóablakos technikával oldja meg, a megbízható átvitelt pedig a sorszámozással és a nyugtázással biztosítja.

Adatok fogadása a viszony rétegtől, Szegmentálás, Szegmensek továbbítása a hálózati rétegnek, Biztosítja a hibamentes átvitelt: Elrejti a felsőbb rétegek elől az átvitel problémáit, Tényleges kommunikáció a végpontok között, Működése hasonló a hálózati réteghez, de Teljes egészében a felhasználó gépén fut.

A 4. rétegbeli protokollok általános jellemzői

A TCP/IP protokollkészlet két, 4. rétegbe (szállítási rétegbe) tartozó protokollja: a *TCP* és az *UDP* protokoll. A *TCP* egy virtuális áramkört hoz létre a végfelhasználói alkalmazások között. A **TCP jellemzői** a következők: összeköttetés-alapú, megbízható, a kimenő üzeneteket szegmensekre bontja, a célállomásnál újra összeállítja az üzeneteket, mindent újraküld, amit a vevő nem tudott venni, a bejövő szegmensekből visszaállítja az üzeneteket.

Az ***UDP* jellemzői**: összeköttetés-mentes, nem megbízható, üzeneteket visz át, nem biztosítja szoftveresen a szegmensek megérkezését (nem megbízható).

A TCP protokoll általános jellemzői

A *TCP protokoll* (Transmission Control Protocol - átvitelvezérlő protokoll) 4. rétegbeli (szállítási rétegbeli) protokoll, amely megbízható, kétirányú (duplex) adatátvitelt biztosít. A TCP protokoll a TCP/IP protokollcsalád része.

A TCP szegmensformátuma

A TCP-szegmensek a következő mezőket tartalmazzák:

***Forrásport** (16bit)*: a hívó port száma

***Célport** (16bit)*: a hívott port száma

***Sorszám** (32bit)*: a megérkező adatok helyes sorrendjét biztosító számmező

***Nyugtaszám** (32bit)*: a következő várt oktett sorszáma

***HLEN** (TCP-fejrészhossz) (4bit)*: a fejrész 32 bites szavakban mért hossza

URG
- Ha értéke "1" ha a sürg_sségi mutató mez_ használt

ACK
- Ha értéke "1" ha a fejrész nyugtát (is) tartalmaz
- Ha értéke "0" a nyugta mez_ átugorható

PSH
- Késedelem nélküli adattovábbítás kérésére

RST
- Összeomlott vagy összezavart összeköttetés helyreállítása

SYN
- Az összeköttetés felépítéséhez használt jelz_bit
- SYN=1 ACK=0 ; Összeköttetés kérés (Connection Request)
- SYN=1 ACK=1 ; Összeköttetés fogadása (Connection Accepted)

FIN
- Az összeköttetés bontásának jelzésére

Fenntartott mező *(6bit)*: nulla értékű

Fódbitek: szabályozási funkciójuk van (pl. az összeköttetés felépítése és bontása)

Ablakméret
- A csúszóablak méretének szabályozása (forgalomszabályzás)
- Lehetséges értékek
    - 0: az adatok rendben megérkeztek, ne küldj további adatokat
    - >0: a nyugtázott bájttól kezd_d_en ennyi bájtot küldj

Ellenőrző összeg
- A teljes szegmens ellenőrző összege

Sürgősségi mutató
- A sürgős adat helyének jelzése az adatok között az aktuális sorszámhoz képest

TCP protokoll

Sorszámozás

- Minden bájt 32 bites egyedi sorszámot kap
- Számlálók átfordulása

TCP szegmens
- Fejrész (20 bájt) + [Fejrész opció] + [Adatok]
- A szegmens méretét a TCP szoftvere határozza meg
    - Több írási m'velet is gy'jthet_ egy szegmensbe, de egy adatsor is küldhet_ több szegmensben
    - Maximum (fejrésszel együtt): 65.515 bájt (IP adatmez_)
    - MTU (Maximum Transfer Unit): gyakorlatban 1.500 bájt (Ethernet adatmező)

TCP nyugtázás

Csúszóablakos protokoll
- Szegmensek küldésekor időzítő indítása
- Szegmens érkezésekor válasz szegmens (nyugta)
    - Tartalmazza a következő várt szegmens sorszámát
    - Felhasználói adatokat (ha van továbbításra váró adat)
- Ha a nyugta nem érkezik meg id_ben ; újraküldés

A nyugtázás lehetséges problémái
- Helytelen sorrendben érkező szegmensek
- Torlódás, nagy késleltetés ; kétszer érkezik a szegmens
- Újraküldésnél eltér_ bájt tartományok a szegmensben
- Mi történjen a nem nyugtázható szegmensekkel?

Az UDP protokoll általános jellemzői

Az *UDP protokoll* (User Datagram Protocol - felhasználói datagram protokoll) a TCP/IP protokollkészlet összeköttetés-mentes átviteli protokollja. Az UDP protokoll egyszerű, datagramokat szállító protokoll, amely nem foglalkozik a nyugtázással, és nem garantálja az átvitelt, azaz a hibakezelést és az újraküldést más protokollokra bízza. Az UDP protokoll nem használ sem ablakkezelést sem nyugtázást, így a megbízhatóságot az alkalmazási szintű protokolloknak kell biztosítaniuk. Az UDP protokollt olyan alkalmazások számára fejlesztették ki, amelyek nem igénylik a szegmensek sorrendhelyes kezelését.

UDP

User Datagram Protocol

Az Internet összeköttetés nélküli szállítási protokollja

UDP szegmens
- 8 bájt fejrész
    - Forrásport (16 bit)
    - Célport (16 bit)
    - UDP szegmens hossz (16 bit)
    - UDP ellen_rz_ összeg (16 bit)
- adatrész

A fejrészben már a portok is megtalálhatók

UDP

Amire az UDP képes
- Interfészt biztosít az IP protokoll használatához
- Párhuzamos kapcsolatok a portok használatával

És amire nem
- Hibakezelés
- Nyugtázás, újraküldés
- Forgalomszabályozás

Jellemző felhasználási terület: kliens-szerver alkalmazások
- Rövid kérések, rövid válaszok
- Ha nincs válasz ; újabb kérés
- Egyszer' megvalósíthatóság
- Kisebb forgalom
- Pl.: DNS

Portszámok leírása

Mind a TCP, mind az UDP protokoll *portszámok* vagy más néven, *csatlakozószámok* (socket number) segítségével kommunikál a felsőbb rétegekkel. A portszámok segítségével tudják megkülönböztetni a hálózaton egy időben folyó "beszélgetéseket". Az RFC1700-as dokumentumban a szoftveralkalmazások fejlesztői abban állapodtak meg, hogy néhány "jól ismert" portszámot fognak bevezetni. (Pl. az FTP programok alaphelyzetben a 21-es portszámot

használják.) Azok az alkalmazások, melyek nem "jól ismert" portszámokat használnak, egy adott tartományból véletlenszerűen kiválasztott portszámot kapnak. A TCP-szegmensben ezek a portszámok adják a forráscím (forrásport), ill. a célcím (célport) értékét. Néhány portot mind a TCP-ben, mind az UDP-ben lefoglaltak, bár meglehet, hogy úgy írták meg az alkalmazásokat, hogy nem használják azokat. A portszámokat a következő szakaszokra osztották:

- A 255 alatti számok nyilvános alkalmazásokhoz tartoznak.
- A 255-től 1023-ig terjedő számokat a piaci alkalmazásokat fejlesztő cégek használhatják.
- Az 1023 feletti tartományra nem vonatkozik szabály.

A végrendszerek a portszámok segítségével választják ki a megfelelő alkalmazást. A forrásportokat a forrásállomás dinamikusan osztja ki. A forrásportok általában 1023 feletti számokat kapnak.

Port	Protokoll	Port	Protokoll	Port	Protokoll
21	FTP	25	SMTP	110	POP3
22	SSH	69	TFTP	143	IMAP
23	Telnet	80	HTTP	161	SNMP

23. tétel

Összeköttetések felépítése és bontása TCP használatával. A TCP pufferelés lehetőségei, problémái és lehetséges megoldásai. A TCP torlódásvédelmének működése, torlódásvédelem az Interneten. Vezeték nélküli hálózatok és a TCP.

Összeköttetések felépítése

- Háromutas kézfogás
- A Server várakozik a bejöv_ kérésekre
- A Kliens csatlakozási kérést küld
  - IP cím + portszám
  - Maximális TCP szegmens mérete
  - SYN=1 ACK=0
- Ha a célgép célportján nincs várakozó folyamat a TCP RST=1 értékkel válaszol
- Ha a Server folyamat elfogadja a kérést nyugtázó szegmenst küld vissza (SYN=1 ACK=1)
- A nyugtát a kliens gép nyugtázza
- Fontos a kezdő sorszám megválasztása

Összeköttetések bontása

- Duplex átvitel
- Bontás irányonként (szimplex átvitel)
- Bontási kérelem FIN=1 (hoszt1)
  - hoszt1 már nem küldhet adatot
  - hoszt1 továbbra is fogadhat adatot hoszt2 felől
- Bontási kérelem FIN=1 (hoszt2)
  - Az összeköttetés vége
- A két hadsereg problémája (támadás = bontás)
  - Elméletben (sem) létezik tökéletes megoldás
  - Gyakorlatban: ha a FIN kérésre nem érkezik időben ACK → a bontást kezdeményező hoszt befejezi a kapcsolatot

TCP pufferelés

Mikor továbbítsa a TCP az alkalmazás által küldött adatokat?
Mikor nyugtázzunk egy átvitelt?
Egyetlen karakter leütése telnet használatával (azonnali továbbítás mellett)

- 21 bájtos szegmenst hoz létre a TCP
- Az IP hozzáteszi a saját fejrészét (41 bájt) - A megérkezett csomag nyugtázása (40 bájt)
- A feldolgozott karakter visszaküldése (41 bájt)
- A megérkezett csomag nyugtázása (40 bájt)
- 1 (2) bájt átviteléhez összesen 162 bájtot használtunk

TCP pufferelés

Nyugták késleltetése 500 ms hosszan

- Ha az alkalmazás feldolgozta a karaktert a nyugta a választ tartalmazó adattal együtt küldhető
- Folyamatos adatfolyam esetén a nyugták megspórolhatók

Bájtonként érkező adatoknál

- A küldő TCP entitás elküldi az első bájtot - Az érkező további bájtokat puffereli az első bájt nyugtájáig
- Majd a pufferelt bájtokat egyetlen szegmensben továbbítja
- Lehetőség a pufferelt adatok küldésére a nyugtázás előtt
- Nem praktikus pl.: távoli asztalon az egér mozgatására

TCP pufferelés

A buta ablak (silly window) jelenség

- A fogadó entitás puferre tele van, nagy szegmensekben kapta az adatokat (ablakméret = 0)
- A fogadó bájtonként olvassa ki a kapott adatokat
- 1 bájt kiolvasása után 1 bájt hely keletkezik a pufferben (ablakméret = 1)
- Egyetlen bájt érkezése után újra megtelik a puffer…

A fogadó akkor küldhessen ablakméret információt

- Ha képes a maximális méret' szegmens fogadására
- Ha a puffere félig kiürült

TCP torlódásvédelem

A torlódásdetektálás

- Az időzítők nyugtázás előtti lejárását okozhatja
  - Zajos, rossz minőségű átviteli közeg
  - Torlódás (az Internet TCP folyamatai ezt feltételezik)

A torlódás okai
- A vevő kapacitása
- A hálózat kapacitása

A TCP eszköze az ablakméret változtatása
- Vevő által szabályozott ablak
- Torlódási ablak
- A kettő minimuma határozza meg a tényleges ablakméretet

TCP torlódásvédelem

Lassú kezdés (slow start) algoritmus
- A torlódási ablak kezd_értékét a maximális szegmens méretre állítják
- Az adó elküld egy maximális szegmenst
- Ha a nyugta időben megérkezik az ablakméret duplázódik
- Az adó az új maximumot kihasználva küld löketet
- Az exponenciális növekedés az első nyugta elvesztéséig tart
- A megelőző érték lesz a megfelelő ablakméret

Torlódásvédelem az Interneten

A torlódási küszöb bevezetése (kezd_érték 64 kB)
Időtúllépés esetén
- A torlódási küszöb új értéke a torlódási ablak fele lesz
- A torlódási ablak új értéke a maximális szegmensméret lesz
- A teljesítőképesség meghatározása a lassú kezdet módosított algoritmusával
  - -Az exponenciális növekedés csak a torlódási küszöbig tart
  - -A torlódási küszöb felett a növekedés csak a maximális szegmensméret értéke (lineáris növekedés)
- A torlódási ablak mérete a vevő ablakméretéig növekszik

Vezeték nélküli hálózatok

A TCP-t megbízható hálózatokra optimalizálták
Kell-e külön törődni a hálózat megbízhatóságával?
- A rétegszemlélet szerint nem (független rétegek)
- A gyakorlatban muszáj …

Alapvető probléma a torlódásvédelem
- A csomagok jellemzően elveszni és nem torlódni fognak
- Az adó sebességének csökkentése csak olaj lenne a tűzre
- A sebességet ilyenkor lehetőség szerint inkább növelni kell

Inhomogén átviteli utak
- ☞ Vezetékes (megbízható) szakaszok
- ☞ Vezeték nélküli (megbízhatatlan) szakaszok

Vezeték nélküli hálózatok

Közvetett TCP
- ☞ A TCP összeköttetés felosztása vezetékes és vezeték nélküli részre
- ☞ Az új összeköttetések határa a bázisállomás lesz

- A bázisállomás mindkét irányba átmásolja a csomagokat
- Két homogén (teljes TCP) összeköttetés
  - -A vezetékes rész időtúllépései lassítják a forrást
  - -A vezeték nélküli rész időtúllépései gyorsítják a forrást
- Hátrány, hogy ha a forrás nyugtát kap a vételről akkor még nem biztos, hogy célba ért a csomag (a bázis nyugtázott)

TCP - torlódásvédelme

- szállítási réteg beli összeköttetés
- összeköttetés alapú összeköttetést valósit meg

[
UDP - összeköttetés mentes
]

- hogy milyen típusú hálózaton bonyorítják
le, az a TCPt nem érdekli
- nincs kitétel
- a torlódást magának a TCPnek kell megoldania
- router hálózati szinten próbál védekezni, de a fõ feladat a TCP-é
- hálózati forgalmat csökkenti le, ha torlódás alakul ki
- nem lehet új csomagot a hálózatra adni, amíg a régi nem ért célba
- az ablakkeretek méretét dinamikusan változtatja
- csak annyi adat kerüljön a hálózatba, amennyit képes kezelni
- torlódás detektálása
- régen a ACK hiányából lehetett következtetni - ma is :)
- úgy kezelõdik az ilyen csomag, mint ami elveszett, de nem feltétlenül
torlódás eredménye
- TCP ezt feltételezi...
- Optikai szálakra cserélik a nagy sebességû trönköket
- zavarvédett, nagy sebességû, megbízható
- kicsi a valószínûségû, hogy csomagvesztés nem csomagvesztés miatt veszik el
- forgalomszabályozás - ablakméret szabályozás - vevõ határozza meg
- torlódásvédelem - ablakméret szabályozás - hálózat szabályozza
- ha a hálózat kezdi kezelni, akkor már régen rossz
- próbálja szabályozni a host implicit információk figyelembe vételével
- figyelje a hálózatba küldött csomagokat, forgalmat, nyugtát
- ha a torlódási ablak kisebb mint a vevõablak mérete, akkor a hálózat szabja meg az adamennyiséget
- a küldõ a torlódási ablak kezdõértékét a legnagyobb használt szegmensméretre állítja be, elküldi és egy idõzítõt indít
- ha a szegmensre nyugtát kap, akkor a torlódási ablak méretét egy szegmens méretével megnöveli, így 2 szegm. méret lesz
a torl. ablakméret
- ha nyugtát kap, növeli a torl. ablak mérete a max. szegmensmérettel...
- ez addig mehet, amíg el nem éri a vevõ max. ablakméretét, vagy idõtúllépés következik be.
- szegmens mérete 1kbyte - torlódást ablak kezdõ értéke... ezek duplázódnak
- 2 4 8 16...
- minden TCP implementációnak támogatni kell

Internet - torlódásvédelem

- algoritmusnak figyelembe kell venni
- az elõzõ két ablakméreten kívül nyilvántart egy torlódás küszöb ablakméretet
- 64 kbyte - dinamikusan változik
- összeköttetés elején
- torlódási küszöb - 64 kbyte
- torlódási ablak - 1 kbyte
- utána
- elküldött szegmens méret:
- 1. 1 kbyte
- 2. 2 kbyte
- 3. 4 kbyte
- 4. 8 kbyte
- idõtúllépésnél, a trol. küszöböt az akt. torlódási ablak felére állítják
- a - kezdõ szegmens méretre
- mindaddig amíg a torlódási küszöböt nem éri el, addig a lassú kezdés algoritmust használják, ha a küszöböt elérik akkor
lineáris növekedést
alkalmaznak az ablak növekedésénél, nem dupláznak, hanem +1 szegmens.
- lefolytó csomagokat ugyan olyan jelentõsséggel bírnak, mintha a nyugta nem érkezett volna vissza, idõtúllépés
- idõtúllépés - implicit jelzés
- folytó csomag - explicit jelzés
- baj van, ha az idõzítõ rosszul van beállítva
- TCP-nél az elsõ idõzítõ - újraadási idõzítõ - RTT idõzítõ
- megmondja, hogy az elküldött csomag nyugtájára mennyi idõt kell várni a
- terjedési idõ elég jól kézben tartható, mondható egy T idõ
- kicsi szórás, elég pontos
- adatkapcsolati rétegben viszonylag könnyû beállítani

- a rétegekben felfelé ez egyre rosszabb
- a hálózati rétegben már nem jósolható meg ilyen jól
- átlag nem igazán jól jósolható, és nagy a szórása
- vagy túl hamar lejár - nagy adatforgalom, torlódás érzékelés
- vagy túl nagy - nem veszem észre a torlódásokat, sokat várok az elveszett
csomagok nyugtájára - csomagkésleltetés megnõ
- dinamikusan kell változtatni az idõzítõt
- a hálózat teljesítõképességét folyamatosan figyelni kell
- RTT aktuális értéke adott
- elõször - jelenleg fennálló állapotokat tekintve a teljes körülfordulási idõ
- utána a host elindít egy idõzítõt, megméri, hogy egy nyugta mennyi idõ múlva ér vissza, és becsüli az új RTT-t:
- RTT_n+1=alfa*RTT_n+(1-alfa)M
- alfa = 7/8 - régi érték nagy súlyozást kap
- jobb: betaRTT, ahol a beta=2
- konstans nem elég rugalmas - finomítani kell
- szórásra is van becslés
- D_n+1=alfa*D_n+(1-alfa)*(RTT_n- M)
- alfa lehet 7/8-ad, de lehet más is
- idõzítés_n+1= RTT_n+4D_n - újraadási idõzítés
- gyorsan, könnyen megvalósítható
- ha az idõzítõ lejárt
- javított algoritmus
- ha az idõzítõ lejárt, akkor az RTT-t megduplázza amíg a csomag át nem ért, és nem a mért érték alapján frissítsük az RTT-t
- életben tartó idõzítõ - TCP sokáig nem észlel forgalmat és lejár a ez az idõzítõ, megnézi, hogy a partner küld-e, ha nincs válasz, automatikusan bontja a kapcsolatot
- folyatódó idõzítõ - prezence - a hostok az adatforgalmat leállíthatják
- az összeköttetés létezik, csak a vevõ leállította
- a vevõ újra akarja indítani a kapcsolatot, de elveszik
- 1, az adó vár a végtelenésig - még nincs kész, vagy a vevõ - mert az adó most nem ér rá
- idõzítõ lejárása után a biztonság kedvéért megkerdezi a vevõt, hogy él-e még a kapcsolat, és fogadhat-e adatot
- erre a vevõ válaszol, az adó megnyugszik :)

24. tétel

Digitális jelek továbbítása analóg csatornán. Az AM, FM és PM modulációk. Multiplexelés. Az Internet néhány vezérlő protokollja: az ARP, RARP, DHCP és ICMP szerepe és működése

Az analóg és a digitális jelek összehasonlítása

A jel egy megfelelő elektromos feszültséget, fénymintázatot, illetve modulált elektromágneses hullámot jelent. Ezek mindegyike képes hálózati adatok átvitelére.

A jelek egyik típusa az *analóg jel.* Az *analóg* jelek jellemzői a következők:

- hullámzó
- feszültsége folyamatosan változik az idő függvényében
- általánosan jellemző a természetben előforduló dolgokra
- több, mint 100 éve széles körben használják a telekommunikációban

A jelek másik típusa a *digitális* jel. A *digitális* jelek jellemzői a következők:

- a feszültség nem folytonosan, hanem ugrásszerűen változik az idő függvényében
- általában a technikai, nem pedig a természetes dolgokra jellemző

A digitális jelek amplitúdója állandó, azonban impulzusuk szélessége, hosszuk (T) és frekvenciájuk változhat. A modern forrásokból származó digitális jelek a *négyszögjellel* modellezhetők, amely ingadozás nélkül, látszólag egy pillanat alatt megy át alacsonyból magas feszültségállapotba.

Digitális jelek előállítása analóg jelek segítségével

Az egyik legjelentősebb matematikai felfedezés Jean Baptiste Fourier nevéhez fűződik. Fourier bizonyította be azt, hogy egy szinuszhullám és annak felharmonikusai eredőjeként bármilyen hullámkép (periodikus jel) előállítható. A felfedezés segítségével ismeri fel a kémfilmekbeli beszédfelismerő készülék a hangokat, és a pacemaker működése is ezen az elven alapul. Az összetett hullámok felépíthetők egyszerű hullámokból.

Egy *négyszögjel* vagy *négyszögimpulzus* felépíthető szinuszhullámok megfelelő kombinációjaként.

Egy bit meghatározása a fizikai átviteli közegen

Az adatátviteli hálózatok egyre nagyobb mértékben függnek a digitális (bináris, kétállapotú) rendszerektől. Az információ alapvető építőeleme a bináris számjegy, melyet bitnek vagy (pongyolán fogalmazva) *impulzusnak* nevezünk. Az elektronikus átviteli közegben a bit a bináris 0 vagy 1 értéknek megfelelő elektromos jel. A 0-s bit egyszerűen nulla voltos feszültségnek, az 1-es bit pedig +5 voltos feszültségnek felel meg, de ennél bonyolultabb kódolás is előfordulhat. A *jelföld* egy fontos fogalom, mely minden feszültség alapú hálózati átviteli közegnél előjön. A megfelelő működés érdekében a *jelföldnek* közel kell lenni a számítógép digitális áramköreihez. Ezt a mérnökök úgy valósították meg, hogy az áramköri lapokra földfelületeket terveztek. A számítógép házát használják közös pontnak, vagyis az áramköri lapok földfelületeit (a *jelföldet*) galvanikusan a házhoz kapcsolják. A *jelföld* biztosítja a jelminták számára a 0 voltos feszültséget.

Optikai jel esetén a bináris 0 alacsony vagy nulla fényintenzitásnak felel meg (sötétség). A bináris 1 pedig nagyobb fényintenzitást (van fény), vagy más, komplex fénymintázatot jelent. Vezeték nélküli jelek esetén a bináris 0 lehet egy rövid hullámlökés, míg a bináris 1 egy hosszabb hullámlökés vagy más, komplex jelminta.

A moduláció és a kódolás fogalma

A kódolás az 1-es és 0-s bitek fizikailag megfogható dologgá való átalakítását jelenti:

- vezetékben haladó elektromos impulzus
- optikai szálban haladó fényimpulzus
- elektromágneses hullámimpulzus a térben.

Az átalakítás két lehetséges módszere: az NRZ kódolás és a Manchester-kódolás.

Az NRZ kódolás a legegyszerűbb. (NRZ = non-return to zero, nullára vissza nem térő). Magas és az alacsony jelszintet használ: az 1-es bitnek általában +5 V vagy +3,3 V, míg a 0-s bitnek 0 V feszültségszint felel meg. Optikai szál esetén az 1-es bitet jelentheti az, hogy egy LED vagy lézerfény világít, míg ha nincs fény, az 0-s bitet jelent. A vezeték nélküli hálózatokban 1-es bit esetén pl. van vivőhullám, míg 0-s bit esetén nincs vivőhullám.

A Manchester kódolás bonyolultabb, de a zajokra érzéketlenebb, és jobban tartja az időzítést. Manchester kódolás esetén a biteket az impulzusok átmenete jelzi, mely rézvezeték esetén feszültségszintekkel, optikai szálaknál a LED fényével vagy lézerfénnyel, vezeték nélküli átvitelnél az elektromágneses hullámok energiájával valósítható meg. A Manchester-kódolásnál a magas-alacsony jelátmenet 1-es bitet, az alacsony-magas jelátmenet pedig 0-s bitet jelent.

További közismert, de bonyolultabb kódolások a következők: NRZI (non-return-to zero inverted, nullára vissza nem térő invertált), a differenciális Manchester-kódolás (a normál Manchester-kódolás egy változata), és a 4B/5B kódolás, amely nem bitenkénti, hanem bitcsoportonkénti kódolást végez. Minden kódolási sémának vannak előnyei és hátrányai.

Amplitúdómoduláció (AM) esetében a vivő szinuszhullám amplitúdóját (magasságát) változtatják (modulálják), és így az amplitúdó hordozza az információt.

Frekvenciamoduláció (FM) esetén a vivőhullám frekvenciájának a változása hordozza az üzenetet.
Fázismoduláció (PM) esetén a hullám fázisa (egy ciklus kezdő és végpontja) változik, és ez hordozza az üzenetet.
A bináris 1-es és 0-s értékek ráültethetők egy vivőhullámra amplitúdómodulációval (hullám be/hullám ki), frekvenciamodulációval (1-es bitnél a vivőhullám nagyon ingadozik, 0-s bitnél kevésbé), illetve fázismodulációval (adott fázisváltás az 1-es és a 0-s bitekhez).
Multiplexelés: a fizikai közeget speciális eszközökkel megosztják több egység között. A multiplexelés során a vonalat meghatározott, rögzített módszer szerint osztjuk fel. Minden bemeneti csatornához tartozik a túloldalon egy kimeneti csatorna is. A vevő oldalon biztosítani kell, hogy az érkező információkat a címzett vegye. Azt a műveletet, amely ezt biztosítja, demultiplexelésnek nevezik. A gyakorlati megvalósítás alapján beszélhetünk frekvencia- és időosztásos multiplexelésről. A frekvenciaosztásos multiplexelés bonyolultnak tűnő, ámde meglehetősen egyszerű vonalmegosztási módszer. Analóg átvitelben használják. Azon a felismerésen alapul, hogy a ténylegesen átvitelre kerülő analóg jelek viszonylag kis frekvenciatartományba esnek. Mivel a vonal sávszélessége ennél jelentősen nagyobb, több ilyen tartomány vihető át egyszerre rajta. Azt kell megoldani, hogy ezek a tartományok egymástól jól elkülöníthetők legyenek. Az analóg jelek esetében megvalósítható az, hogy a kisfrekvenciás jelek ráültethetők egy nagyobb frekvenciájú jelre. A vevőoldalon ezt a jelet kivéve az eredeti analóg jelsorozat rendelkezésre áll. Azt a jelet, amelyre az információt hordozó analóg jeleket rákeverik, vivőjelnek, vagy vivőfrekvenciának nevezik. Az adó oldalon a csatornák jeleit ráültetik egy-egy vivőfrekvenciára (modulálják). Ezeket összegzik, majd a jelek összegét átviszik a vevő oldalra. Ott a jeleket szűrőkkel szétválasztják, majd egy második szűrés során a hasznos jel alól kiszedik a vivőjelet.

A RARP protokoll (Reverse Address Resolution Protocol - fordított címfeloldási protokoll) IP-címeket rendel a MAC-címekhez. Néhány hálózati készülék ugyanis csak e hozzárendelés segítségével tudja az adatokat beágyazni, majd kiküldeni a hálózatra. Itt elsősorban a diszk nélküli munkaállomásokra, ill. a nem intelligens terminálokra kell gondolnunk, mivel ezek a saját MAC-címüket ugyan ismerik, de IP-címüket már általában nem. A RARP protokoll használatához egy RARP-kiszolgálónak kell működni a hálózaton, mely a RARP- kérésekre válaszol.
Tegyük fel, hogy egy forrásállomás, mely a saját MAC-címét ismeri, de IP-címét nem találja ARP-táblájában, adatokat akar küldeni egy másik készüléknek! Ahhoz azonban, hogy a célállomás megkaphassa, és az OSI modell felsőbb rétegeinek továbbíthassa az adatokat, majd válaszolhasson a küldőnek, a forrásállomásnak a saját MAC-címét és IP-címét is meg kell adnia. Ezért a forrásállomásnak saját IP-címe kiderítéséhez ki kell adnia egy RARP-kérést. Ezért a készülék összeállít egy RARP-kéréscsomagot, és kiküldi azt a hálózatra. A RARP-kérést
az üzenetszórási (vagy röviden szórási) IP-címre küldi, hogy azt a hálózat összes készüléke lássa.
A RARP-kérés egy MAC-fejrészből, egy IP-fejrészből és egy ARP-kérés üzenetből áll. A RARP csomagformátuma helyet biztosít mind a forrás-, mind a célállomás MAC-címének. A kérő állomás (a forrás) saját IP-címének mezőjét üresen hagyja. Mivel a RARP-üzenetet a hálózaton lévő valamennyi készüléknek el akarja küldeni, ezért csupa bináris 1-et állít be a cél IP-címének. A RARP-ot használó munkaállomások ROM- ban tárolt program segítségével indítják el a RARP-folyamatot, illetve találják meg a RARP-kiszolgálót.
A DHCP protokoll működéséhez a DHCP-kiszolgálónak egy kioszthatő IP-címhalmazzal (címtérrel) kell rendelkeznie. Amikor egy hálózatra kötött állomást bekapcsolnak, az állomás felveszi a kapcsolatot a DHCP-kiszolgálóval, és kér egy IP-címet. Ekkor a DHCP-kiszolgáló választ egy címet, és lefoglalja azt az állomás számára. DHCP használatával a számítógép az összes beállítást megkaphatja egyetlen csomagban. (Például az
IP-cím mellett a kiszolgáló elküldheti az alhálózati maszkot is.)
Az IP egyik fő eleme az **ICMP protokoll** (Internet Control Message Protocol - internet vezérlőüzenet protokoll). Ezt a protokollt a készülékek arra használják, hogy az üzenet küldőjét értesítsék az esetleges hibákról. Ha
például egy forgalomirányító olyan csomagot kap, amelyet nem tud kézbesíteni, erről értesíti a csomag küldőjét.
Az ICMP protokoll egyik része az ún. *visszhang-kérés/visszhang-válasz* (echo-request / echo-reply), mely a célállomás ún. *pingelését* teszi lehetővé, azaz annak eldöntését, hogy a csomagok képesek-e elérni a célállomást.
Az ARP protokoll
A keresett MAC-címet, amelyet a beágyazott adatokhoz kell csatolni, a készülékek sokféleképpen meg tudják határozni. Lehet például az adott LAN-ra kapcsolódó összes készülék MAC- és IP-címét táblákban tárolni.
Ezeket a táblákat az *ARP protokoll* (Address Resolution Protocol - címmeghatározó protokoll) használja, ezért
ARP-tábláknak hívják őket. Ezek a táblák az összetartozó IP-cím-MAC-cím párokat tárolják. Az ARP-táblákat RAM memóriában tárolják, melyet minden készülék maga tart karban. Az ARP-tábla bejegyzéseit gyakorlatilag sohasem kell kézzel bevinni. A hálózat minden számítógépe maga tartja karban az ARP-tábláját. Amikor egy hálózati készülék adatokat akar a hálózaton átküldeni, ezt az ARP-táblájában található információ alapján teszi. Miután a forrás meghatározta a cél IP-címét, megkeresi a hozzá tartozó MAC-címet az ARP-táblában. Ha van az
IP-címnek megfelelő bejegyzés a táblában (vagyis az IP-célcímhez tartozik MAC-cím), akkor a MAC-címet az
IP-címhez rendeli, és ezt használja az adatbeágyazáskor. Ezután az adatcsomag már kiadható a hálózati átviteli közegre, hogy a cél megkaphassa.