Az Internet

Az Internet kialakulása

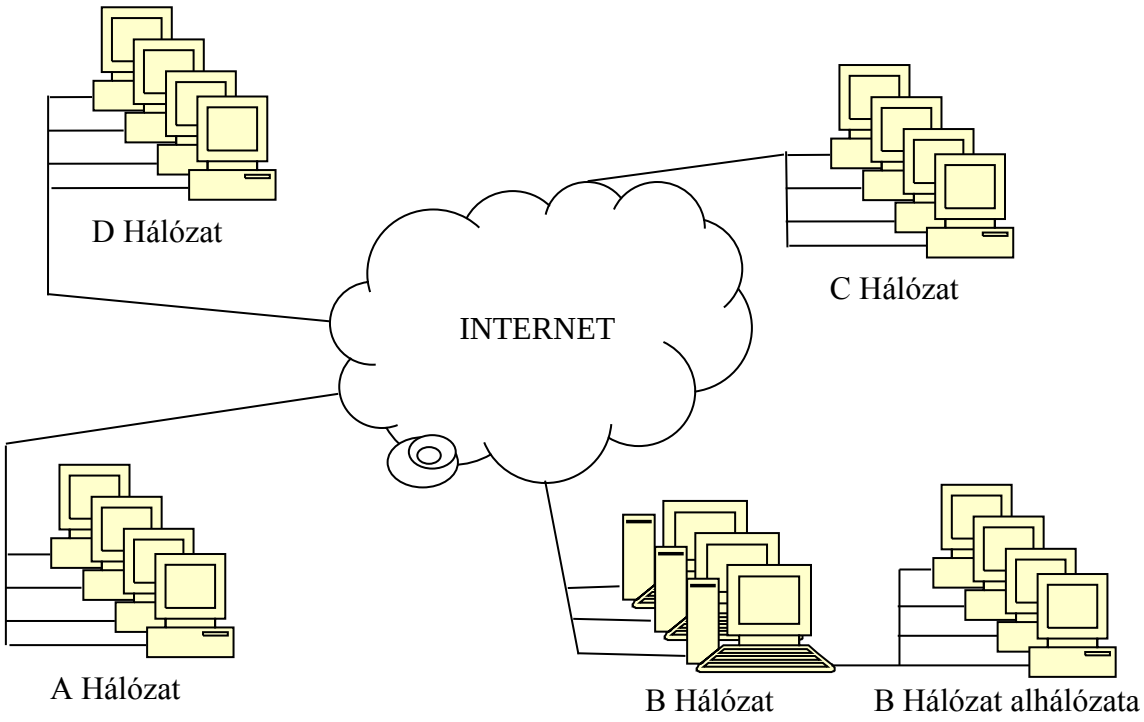
Az ötvenes-hatvanas években merült föl az USA-ban egy kevéssé sebezhető számítógép-hálózat szükségessége, amelynek egy esetleges atomtámadás után megmaradó részei működőképesek maradnak. 1957-ben Eisenhower elnök elrendelte a Defence Advanced Research Project Agency (DARPA) létrehozását, amely a kutatásokat finanszírozta. A kutatások során egy több központú, csomagkapcsolt (ahol az adatok továbbítása kisebb egységekben történik) hálózati kommunikációs protokollt dolgoztak ki, amely a mai TCP/IP őse volt. Ezen az elven alapult az 1969-ben elindított ARPANET hálózat. Az ARPANET-et a katonai felhasználásokon kívül egyetemek, kutató intézetek is használni kezdték a csomagkapcsolt protokollal történő adattovábbítás kutatásának céljából. A kutatás eredményeként elkezdték használni a tudósok üzenetküldésre (1972: első e-mail program megjelenése), fájlok cseréjére is. 1974-ben egy TCP protokollról szóló tanulmányban először használták a csomagkapcsolt hálózatra az INTERNET kifejezést. 1983-ban az USA kormánya szétválasztotta a katonai és a polgári célú hálózatfejlesztést. Az ARPANET lett a polgári célú hálózat, a MILNET nevet pedig a hadsereg hálózata kapta. Ezzel a lépéssel született meg a mai INTERNET. A National Science Foundation véleménye szerint a hálózat nagyon fontos lehet a későbbi kutatásokban, ezért célként tűzte ki maga elé az Internet jelentős bővítését. 1985-ben egyik legjelentősebb lépésként létrehozták az NSF6 szuper-számítógép központjukat és kialakítottak egy saját hálózatot (NSFNET) is. 1986-ban az NSFNET-et összekapcsolták az ARPANET-tel. Az NSFNET ezek után tovább folytatta a fejlesztéseket. Újabb számítógép központokat hozott létre, melyeket már optikai kábelekkel kötött össze. Az akkor kialakított optikai hálózat mai napig az USA egyik legjelentősebb gerinchálózata, mely 56 kbit/s sebességgel kezdte meg működését, de manapság már több gigabit/s sebességű. 1989-ben az ARPANET megszűnt, helyet adva a modernebb hálózatoknak. Ebben az időben több magáncég (AT&T, UUNET, MCI) is épített ki saját hálózatot és csatlakozott az NSF hálózatához, vagyis az Internethez, látva az üzleti lehetőségeket. Az azóta eltelt években több ezer különálló hálózat sok tízezer számítógépét kapcsolták a folyamatosan növekvő internethez.

A 80-as évek végén az NSFNET-hez hasonló elvek alapján számos országban szerveződtek gerinchálózatok. Ezek mindenekelőtt az NSFNET-hez csatlakoztak, de gyakran egymással is kiépítették közvetlen kapcsolatokat.

Az internet talán legfontosabb szervező, összefogó ereje az Internet Society (ISOC). A társaság nyílt, tagja lehet bármely szervezet vagy magánszemély. Célja az internetes technológiával történő információcsere összehangolása, fejlesztése és a hálózat működéséhez elengedhetetlen Internet Protocol (IP) fejlesztése. Az ISOC által felkért, nagy szakmai tekintéllyel rendelkező önkéntesekből áll az Internet Architecture Board (IAB) melynek feladata hogy állást foglaljon alapvető stratégiai kérdésekben, felelős a szabványok elfogadásáért, ill. a szabványosítást igénylő kérdések meghatározásáért és a címzési rendszer karbantartásáért.

Az internet felépítése

Hálózatról akkor beszélünk, ha legalább két számítógépet valamilyen módon (manapság leggyakrabban Ethernet hálózattal) összekötünk. A kialakítás topológiája alapján el tudunk különíteni hálózatot és annak alhálózatát. Az internet sok ilyen hálózat összekapcsolásából jött létre.



Az internet felépítése

Működés

Az internet teljes protokollja négy rétegből álló, de az ISO:OSI bizottsági hálózati modellnél egyszerűbb kialakítású. Minden rétegnek megvan a maga feladata, amit végrehajt vagy amiért felelős, és az információt továbbadja a következő rétegnek.

Fizikai és adatkapcsolati réteg

A legalsó réteg egyszerűen csak keretezett adatokat juttat el A pontból B pontba. A hálózatban A pontnak és B pontnak saját címe van (ez az egyedi MAC cím), ami alapján azonosíthatók. Például, ha Ethernet hálózatot használunk, akkor az egyes számítógépeket az Ethernet cím azonosítja (mely teljesen egyedi), így ha A gép üzenetet akar küldeni a B gépnek Etherneten át, akkor vagy mindenkinek küld (broadcast), vagy ismeri B gép Ethernet címét.

IP (hálózati) **réteg**

A fizikai rétegre épül az **IP csomagküldő réteg, mely a csomag megfelelő útvonalon történő célcímre juttatását végzi**. Ez a kapott adatokat csomagokba szervezi, majd ezeket átadja az alatta lévő rétegnek továbbításra. **A forrás- és a célgépet az IP cím azonosítja.** Ez (IPV4 esetén) 4 bájtos (32 bit), mely **két részre bontható (**Pl: 201.121.34.12, az egyes ponttal elválasztott decimális számok egy-egy 8 bites részt jelentenek a 32 bitből):

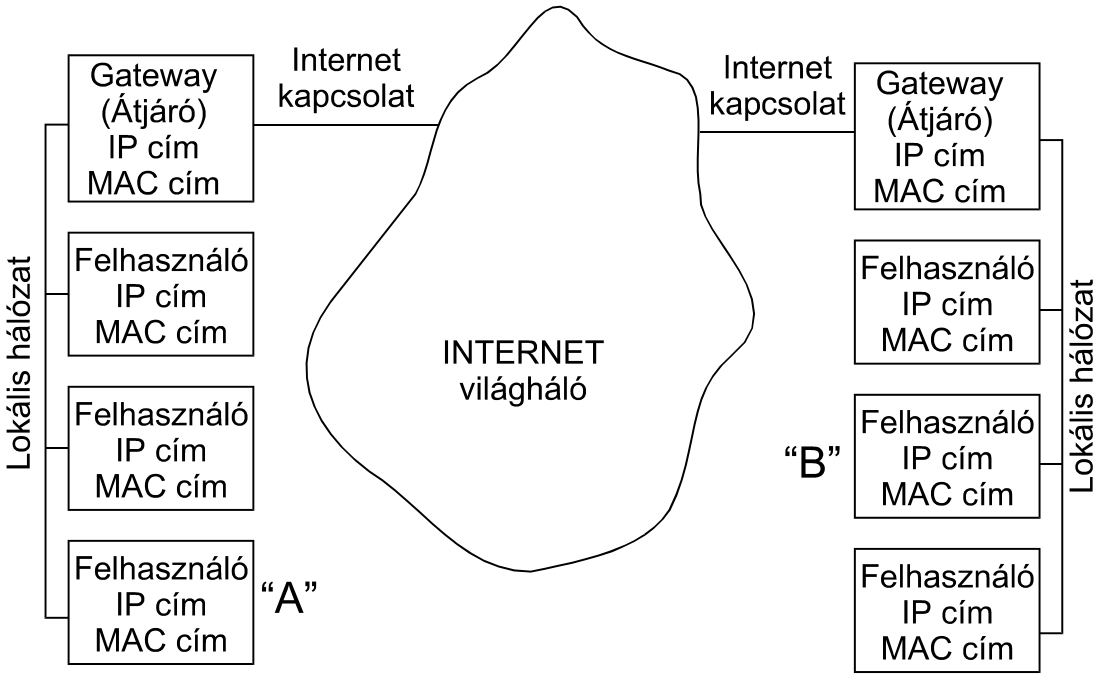
1. **Hálózat-azonosító. (**Pl: 201.121.34.12-ben a pirossal jelölt rész)
2. **Gépazonosító a hálózaton belül.** (Ha n bites, akkor 2^n-en gép lehet a hálózaton belül) **(**Pl: 201.121.34.12-ben a zölddel jelölt rész**):**

A két mező aránya változó.
A két mezőt a 32 bites (IPV4)**, úgynevezett alhálózati maszk különbözteti meg.** Ezt byte-onként decimálisan szokták megadni, **a hálózat azonosító bit-jeit a maszkban bináris 1-ek jelölik. A mask bitjei balról indulva csupa 1-esekből állnak, ahol a maszk bitek 0-ák, azok a bitek már a gépazonosítóhoz tartoznak.** Az alhálózati maszkot decimális számokkal jelölik. Itt a könnyebb érthetőség végett elöször hexadecimális számokkal jelölve, például az hexa **FF FF FF** 00 alhálózati mask decimálisan: **255.255.255.0**. Ez azt jelenti, hogy a hálózat címe 24 bites (3 byte-os), a hálózaton belüli cím (gép azonosító) 8 bites (1 byte-os). Tehát 256 gép különböztethető meg. Másik példa: alhálózati mask hexa FF FF FF C0 decimálisan 255.255.255.192 esetén (a gép azonosító annyi bites, amennyi a mask végén a 0 értékű bitek száma, itt 6) tehát ebben a hálózatban 2^6= 64 db gép különböztethető meg.

Amikor az A számítógép csomagot küld B számítógépnek, első lépésben eldönti, hogy ugyanabban a lokális hálózatban vannak-e. (Az IP címének hálózat azonosító része megyezik-e a sajátjával.) Ha igen, akkor egyenesen B-nek fog üzenni, ha nem, akkor az **hálózati átjáróként (gateway)** definiált gépnek (és a csomag célba juttatásáért a továbbiakban az a felelős).
Az Internet Protokollban A gép B-nek csak az IP címét ismeri, ám mivel a **lokális hálózatban** általában a **MAC cím alapján lehet csak küldeni**, így a **küldés előtt ki kell találni az IP címhez tartozó MAC címet is**. Erre való az ARP (Address Resolution Protocol). A gép kiküld egy **broadcast (mindenkinek szóló) üzenetet**, hogy keresi a megadott IP-című gépet. B gép rájön, hogy ez ő, és visszaüzen A-nak. Ettől kezdve A ismeri B gép MAC címét, és küldheti a csomagot.

Az átjáró (gateway) feladata az, hogy a kapott csomagot eljuttassa a megfelelő hálózat átjárójához. Onnantól a célba juttatás már megoldott (ARP-vel). Ehhez különféle útvonal-választó algoritmusokat használ. Az egész alapja az, hogy **minden útvonal-választó gépnek (router) van egy táblázata, hogy melyik hálózat merre van**. Ezeket folyamatosan frissítik, és egymással is egyeztetik. Ha minden router csak azt tudja, hogy neki merre kell a csomagot továbbküldeni, így az előbb-utóbb eléri a címzett hálózatot (majd azon belül a címzett gépet).

Példa



A gép IP címe: 201.121.34.12 [Alhálózati maszk: 255.255.255.0]
B gép IP címe: 201.121.35.76 [Alhálózati maszk: 255.255.255.0]

A gép el akar küldeni B gépnek egy csomagot, de csak az IP címét ismeri. Lépések:

1. ***A*** gép és ***B*** gép ugyanabban a hálózatban vannak? Az alhálózati maszk alapján az IP cím első 3 száma adja meg a helyi hálózatot. Ez eltér, tehát ***A*** és ***B*** nincsenek ugyanabban a helyi hálózatban.
2. Ekkor tehát a csomagot a Gateway-nek kell elküldeni. Ennek IP címét a hálózaton belüli gépek ismerik. Most: 201.121.34.1.
3. Mivel az Ethernet hálózat csak Ethernet-kártya azonosító alapján tud küldeni, kell a Gateway Ethernet-címe is (MAC).
4. ***A*** gép kiküld egy broadcast (mindenkinek szóló) üzenetet: „Kinek az IP címe a 201.121.34.1? (Másképp: mi a MAC címe a kiküldött IP című gépnek?)
5. Erre egy gép (a Gateway) válaszol: 00-0C-6E-3A-BE-9A jelzi, hogy ő az.
6. Ekkor ***A*** elküldi a csomagot 00-0C-6E-3A-BE-9A-nak (tehát a Gatewaynak).
7. A Gateway megnézi a táblázatában, hogy hol van a 201.121.35-ös hálózat, a cél gép hálózata. Majd átküldi a csomagot a 201.121.35.1-es gépnek, az ottani Gateway-nek.
8. A 201.121.35.1-es ottani Gateway is megnézi, hogy B gép az ő hálózatában van-e. Igen, mert az alhálózati címe az övével egyezik (201.121.35).
9. Természetesen megint csak szükség van B MAC-címére. Ismét küldeni kellene a kérdést, várni a választ, de a Gateway-ek gyakran belső táblázatokban tárolják az IP-MAC párosítást, így megspórolható a kérdés: a táblázat szerint B gép 00-0D-7E-3D-11-23
10. A 201.121.35.1-es Gateway elküldi a csomagot 00-0D-7E-3D-11-23-nek, a B gépnek, amely ezzel célba is ért.

TCP és UDP (szállítási) réteg

Az IP-csomagok gyorsan és hatékonyan továbbíthatók, de semmilyen garancia nincs arra, hogy célba érnek és a sorrendjük is jó lesz. Ezért közvetlenül ritkán dolgoznak IP csomagokkal, inkább a ráépülő protokollokat használják.

TCP

A TCP az IP-rétegre épülő összeköttetés-alapú adatátviteli protokoll (visszajelzés jön a csomag megérkezéséről). **Hibamentes adatfolyam-átvitelt biztosít.**
Az **adatfolyamot ehhez elsőként darabokra bontja**, s így alakulnak ki a **csomagok**. Ezeket utána IP csomagokként elküldi. A kapcsolat hibamentes átvitelt garantál (hiba detektálás és csomag újraküldés). **Minden csomag sorszámozott, és mindegyikre választ vár a küldő.** Ha valamelyik csomag elvész vagy megsérül, akkor gondoskodhat a hiba javításáról. Így a további rétegeknek már nem kell ilyenekkel foglalkozniuk: biztosak lehetnek benne, hogy a bájtfolyam, amit elküldtek, ugyanígy is fog megérkezni.
Sajnos a hibamentességnek ára van: a **TCP protokoll lassú, nehézkes működésű és bonyolult**.

UDP

Az UDP szintén az **IP-rétegre épülő** datagram-küldő protokoll (nem kap visszajelzést a célbaérkezésről). **Adatcsomagokat küld, de nem vállal semmilyen garanciát, hogy azok meg is fognak érkezni.** Cserébe **gyors és egyszerű**, így pl. on-line rádiók esetében sokkal előnyösebb.

További protokollok

A TCP-re és az UDP-re számos további protokoll épül. A legismertebbek:

1. **FTP. (File Transfer Protocol).** A TCP-re épülve fájlokat visz át a hálózaton. Nem biztonságos átviteli forma, helyette az SCP használatos.
2. **http** (Hyper Text Transfer Protocol). **A WEB böngészés alap-protokollja.** Alapvetően fájlokat visz át (mint az FTP), de sokkal specifikusabb (pl. támogatja a frissítést, stb.)
3. **SMTP** (Simple Mail Transfer Protocol). **Elektronikus levélküldés protokollja.** Az E-mail-eket a routerek egymásnak adogatva juttatják el a címzetthez.
4. TELNET, SSH, stb. Távoli terminál.

Alkalmazási réteg

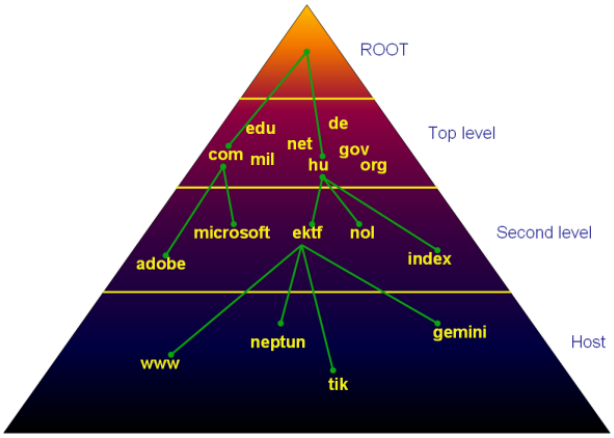
A legfelső réteg az alkalmazási réteg. Itt már maga az információ jelenik, amit a különböző alkalmazások már fel tudnak dolgozni.

DNS (Domain Name Server)

Az Interneten a gépeket az IP-címük azonosítja, ám ennek megjegyzése nehéz feladat. Ezért inkább **szöveges neveket használunk.**
Pl.:

www.index.hu = 217.20.131.2

A működéshez természetesen a beírt nevet előbb IP címmé kell alakítani. **A szöveges név és az IP cím összepárosítását a DNS (Domain Name Server) végzi.**
A feladat lényegében egy egyszerű táblázat kezelése, ám a méretek és a frissítés problémái miatt a feladatot decentralizáltan oldották meg.
A rendszer hierarchikus, **fa szerkezetű.** Az egyes hierarchia-szintekre utal a nevek bontása is: **a név ponttal elválasztott mezőkre, domainekre oszlik** (*gépnév.aldomain.domain, pl. neptun.bme.hu).*



Minden hálózatban van egy vagy több DNS szerver számítógép. Ennek IP címét minden helyi számítógép ismeri. A helyi DNS ismeri a hierarchiában felette álló DNS IP-címét. Az is ismeri a felette lévő DNS IP címét, valamint az összes alárendelt DNS IP-címét is.

Ha fel kell oldani egy (ismeretlen) nevet, akkor elsőként a helyi DNS kapja meg a feladatot:

„Ki az a www.index.hu?”

A helyi DNS csak a helyi hálózat neveit ismeri, **ha keresett gép nincs az adatbázisában, akkor a kérést továbbítja a felsőbb szintű DNS-nek**:

„Kérdezik tőlem, hogy ki az a www.index.hu, de nem tudom. Ki tudhatja?”

Ha a felsőbb szintű DNS tudja, hogy melyik alárendelt hálózatról van szó, akkor oda továbbítja a kérdést, ellenkező esetben ismét csak felfelé.

„A keresett gép címe 217.20.131.2”

A legfelsőbb szintű domaineket központilag menedzselik. A domain-neveknek komoly kereskedelmi- és marketing szerepe van, így ezek kereskedelmi termékek. Ma is komoly csatározás zajlik a legfőbb domain-nevek körül.

Címfordítás

Vannak olyan routerek-ek, melyek képesek címfordítást végezni. Ilyenkor egy teljes helyi hálózat egyetlen IP-címen jelenik meg az Interneten, és úgy tűnik, mintha a helyi IP-hálózat (az Intranet) gépein futó programok mind a routeren futnának. **Így egy nagyobb hálózat „eldugható” egyetlen IP-cím mögé.**

Az Intranet gépei ilyenkor speciális, csak az adott helyi hálózatban „élő” címeket kapnak. Ezeket a router „fordítja” át a „valós” IP-címre.

A 192.168… IP címek jellemzően ilyen címeket jelentenek.

A megoldás előnye, hogy IP-cím takarékos. Hátránya, hogy csak belülről kifelé lehet kapcsolatot kezdeményezni, mivel az Intranet gépei kívülről nem különböztethetők meg (illetve csak korlátozásokkal).

Proxy

A Proxy szerver, „Helyettes” egy olyan gép, mely eljátssza, hogy ő a címzett szerver. A háttérben begyűjti a kért adatokat a címzett szervertől, majd a címzett szerver nevében válaszol.

Alapvetően gyorsítási célra használják: a Proxy szerver folyamatosan tárolja (menti) pl. a letöltött web-lapokat, így a következő lekérdezésnél a lassabb távoli elérés helyett gyorsan válaszol.

DHCP

(Dynamic Host Configuration Protocol)

Minden IP-hálózatba kötött gépnél meg kell adni a következő adatokat:

1. **IP cím**
2. **Alhálózati maszk** (ez dönti el, hogy az IP-címnek mely része a hálózati azonosító, és melyik a gépazonosító)
3. **DNS IP címe** (általában több is megadható).
4. **Átjáró (Gateway) neve vagy IP-címe.**

Vannak további beállítások is, melyek hálózat-specifikusak.

Mindez egy laikusnak bonyolult és zavaros, ráadásul hibalehetőségeket rejt magában.

A DHCP lehetőséget teremt arra, hogy ezeket az adatokat a számítógép egy központi szervertől kérje le (beállítja magát a gép). Így a felhasználónak nem kell az IP rendszer beállításával foglalkoznia, és a gép központilag menedzselhető.

TCP ütemezés

A Internet egy sarkalatos alapelve az igazságosság: senki sem éheztetheti ki a másikat. Ugyanakkor a lehető legnagyobb sebességgel akarunk átvinni a hálózaton. Ez az ellentmondás a TCP protokollnál jelenik meg a legmarkánsabban.
A TCP (mint szó volt róla) adatfolyamot visz át a hálózaton. Ehhez az adatfolyamot csomagokra bontja, majd a csomagokat elkezdi átküldeni az Interneten. Mindegyikre választ vár, de persze semmi akadálya annak, hogy a válasz megérkezése előtt már elküldje a következő csomagot. Ezzel a késletetési idő csökkenthető, így gyorsítható az átvitel.
Természetesen, ha a gép 10 csomagot küldött el, akkor 10 nyugtázó válasznak is kell majd érkeznie.
Mindez azt jelenti, hogy a hálózaton egyidőben több csomag és visszaérkező nyugta is utazhat. A kérdés az, hogy mennyi? Ha kevés csomag utazhat csak egyidőben, akkor lassú lesz az átvitel. Ha túl sokat küldünk ki, akkor eltömíthetjük a csatornát, esetleg kizárva mások kommunikációját is.
Mivel az Internet nem homogén, elképzelhető, hogy a csomagok útvonalán van egy szűk keresztmetszet, ahol torlódás lép fel. **A torlódásnál** a router egy darabig tárolja a küldendő üzeneteket, de ha a puffere betelik, akkor elkezdi eldobálni a csomagokat: tehát **csomagvesztés lesz** (vagy az adatcsomag, vagy a nyugta elvész).
Ez egy indikátora annak, hogy túl gyorsan akarunk adatokat küldeni, s ezt a TCP ki is használja.
A mechanizmus eléggé bonyolult, itt nincs mód a pontos ismertetésére.
A működés röviden a következő:

1. Elkezdi küldözgetni a csomagokat**. Kevés csomagot küld nyugta nélkül, közben várja a választ.**
2. **Ha minden rendben, megjött minden válasz, akkor gyorsít, és kicsivel több csomagot küld el nyugta nélkül.**
3. Ha továbbra is minden OK, akkor még több csomagot küld el anélkül, hogy várna a nyugtára.
4. Egy idő után valahol telít a hálózat, és hirtelen elkezdenek eltűnni a csomagok.
5. **Ha csomagvesztés van, akkor drasztikusan visszavesz a sebességből, tehát ismét csak kevesebb csomagot küld ki nyugta nélkül, és kezdi az egészet elölről.**
6. Természetesen a hibákat javítani kell.

Csomagszeletelés

Egyes hálózat-fajták (ilyen az Ethernet is) felülről **limitálják az átvihető adatcsomagok méretét.** Tehát **előfordulhat, hogy az átviendő IP csomag nem vihető át egyben**.
Ezt a küldő természetesen nem tudhatta. Így pl. a routernek kell megoldani egy 32kbyte méretű IP csomag átvitelét egy 1350 bájt maximális csomagméretű hálózaton.
Ezt **a probléma megoldása, hogy lehetőség van a csomagok szétbontására, majd a fogadó oldalon az újraegyesítésre.**

IPv6

A IP címek a jelenleg használt IPv4-es rendszerben 32 bitesek. Ez 4 milliárd számítógép számára lenne elég, ám kezdetben ez olyan soknak tűnt, hogy elég bőkezűen osztogatták a címeket. Így **mára erősen fogyóban van a szabad IP-cím.**
Az IPv6-os rendszer egyik nagy erőssége a 128 bites IP cím, ami most tűnik nagyon soknak…
Az új protokoll további korszerűsítéseket is tartalmaz, ám az inkompatibilitási problémák miatt az IPv6 csak lassan terjed.

Szolgáltatások

World Wide Web - Honlapok működése

A http protokoll eredetileg elektronikus dokumentumok publikálására született. Ez azon is látszik, hogy egy korabeli nyomdai szabványt használt fel a dokumentumok reprezentálására. A kialakult hyper-szöveg szerepe viszont lassan eltolódott, és több más internetes technológiát háttérbe szorítva vagy magába olvasztva lassan az Internet egyik legjelentősebb képviselőjévé vált.
A WEB alapműködése szerint **a szerveren fut egy WEB-szerver program, mely válaszol a böngésző lekérdezéseire.** A **http (h**yper**t**ext **t**ransfer **p**rotokoll**)** alapvetően egy nem túl bonyolult **fájl-átviteli protokoll**, melynek segítségével **a megfelelő könyvtárban lévő fájlokat küldi vissza, melyek aztán a böngészőben dokumentumként jelennek meg.** (Pl. **html** file, HyperText Markup Language).
A technológia eléggé statikus volt (a megjelenített web-lapon csak kézzel lehetett módosítani), amit elsőként a CGI tört meg**.** A CGI egy olyan program, mely a **lekérdezéskor dinamikusan állítja elő az elküldendő HTML fájlt**, melybe azt ír, amit akar**.** Így viszonylag egyszerűen **létrehozhatók interaktív dinamikus weboldalak.**
Később megjelentek a böngészőben futó egyszerű szkriptek és programocskák (appletek), valamint ezek szerveren futó változatai (szervletek). Ezzel a statikus WEB végleg háttérbe szorult. Ezzel párhuzamosan megjelent a titkosított átvitel (főleg banki tranzakciókhoz), és az elektronikus kereskedelem.

E-mail

Elektronikus levelezés. Eredetileg a file-átvitel speciális eseteként egyszerű, az angol ábécé betűit tartalmazó szövegfile-ok továbbítását jelentette. Mára már sokkal bővebb szolgáltatásokat nyújt: bővítmények (pl. MIME) segítségével tetszőleges karakterkészlettel írt szövegek, képek, hangok, mozgóképek, ill. mindenféle bináris file-ok is átvihetők. **Az e-mail-ek továbbítását az SMTP (Simple Mail Transfer Protocol) protokoll végzi.** Egy **központi szerveren elhelyezkedő levelezés távoli elérésére szolgálnak** a **POP3** és az **IMAP protokollok** (a POP3 egyszerű levélletöltést tesz lehetővé otthoni gépre minimális idejű hálózathasználattal, míg az IMAP-pal az e-mail-eket magán a szerveren tudjuk rendezgetni, olvasgatni, állandó hálózati kapcsolat mellett).

File-átvitel

File-ok átvitelére alapesetben az FTP (File Transfer Protocol) egyszerű és nem biztonságos protokollt alkalmazzák, ehhez úgynevezett FTP szervert kell létrehozni.
Egy gép egyes könyvtárjait a hálózat felől láthatóvá lehet tenni.

Keresőgépek

Az Internet növekedésével új probléma jelent meg: a hatalmas méretű hálózatban igen nehezen lehetett megtalálni a felhasználót érdeklő információt.
Ekkor a DEC egy új számítógépének reklámozásaként új szolgáltatást vezetett be: az internetes keresést. Az egész célja csak az Alpha 8400 TurboLaser rendszer tudásának reprezentálása volt, ám keresés hamarosan komoly iparággá nőtte ki magát.
Az elv egyszerű: az Interneten a linkek mentén automata robotok (adatgyűjtő programok) bolyonganak a web-en, és információt gyűjtenek. Belső szervereken tárolják vagy a teljes oldalt, vagy annak egy kivonatát, majd továbblépnek a linkek mentén. A jelenlegi legnagyobb, a **Google lényegében a teljes Internetet folyamatosan lemásolja belső adatbázisába, majd kereséskor ebből dolgozik.**

Az igazi kérdés a keresőgépek piacán ma a relevánsság: a keresésre kapott információ mennyire hasznos a felhasználónak? Mennyire olyan eredményeket ad a keresés, ami a felhasználó célja volt? Természetesen a nagy cégek a keresési és szűrési algoritmust ipari titokként kezelik.
Minél relevánsabb egy keresőgép, annál elégedettebbek a felhasználók, annál többen keresgélnek az adott keresőgéppel. Az eredmények közé természetesen fizetett hirdetések kerülnek (jó esetben szeparáltan), ezek adják a keresőcégnek a bevételt (jó esetben).

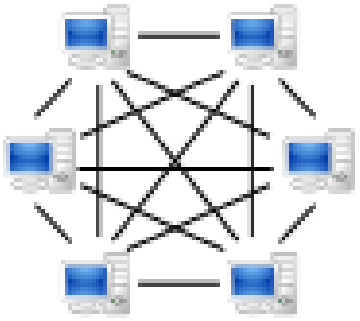
Titkosítás, Digitális aláírás

Itt csak egy rövid áttekintésre van lehetőség.
Minden klasszikus titkosítási eljárásnak van egy közös problémája: a kibontáshoz szükséges kódkulcsot valahogyan el kell juttatni a célállomáshoz, ahol azt ráadásul még tárolni is kell. Ezt a problémát oldja meg a nyilvános kulcsú titkosítás, mely forradalmasította az internetes titkosítási és hitelesítési technikákat.
A **nyilvános kulcsú titkosítás** szemléltetésére **képzeljünk el egy olyan lakatot, amit bárki be tud zárni, de a kinyitni csak kulccsal lehet.**
Ha azt akarom, hogy titkos üzeneteket lehessen nekem küldeni, egyszerűen csak vásárolok egy csomó ilyen lakatot. Mindegyiket kinyitom, a kulcsokat zsebre teszem, a lakatokat pedig szétküldöm az ismerőseimnek. Ha üzenni akarnak, az üzenetet beteszik egy dobozba, a dobozt lezárják a lakattal, majd elküldik nekem. Kinyitni csak én tudom, a kulcs viszont sosem ment át nem megbízható csatornán, nem adtam oda senkinek, abszolút biztonságban volt.
Ezen az elven működik a nyilvános kulcsú titkosítás is.
Kerestek egy olyan matematikai műveletet, mely az egyik irányba egyszerűen végrehajtható, a másik irányba viszont rendkívül nehéz. Ilyen például a prímösszetevőkre bontás. **Két hatalmas (több száz számjegyes) prímszámot összeszorozni egyszerű, de a szorzatból kitalálni a két prímszámot igen nehéz feladat.**
Márpedig az RSA algoritmus esetében *a titkosításhoz elég a két szám szorzata*, a *kibontáshoz viszont már az összetevőkre is szükség van*.
Ha tehát azt akarom, hogy bárki küldhessem nekem titkos üzenetet, kitalálok két hatalmas prímszámot, ezeket titokban tartom, de publikussá teszem szorzatukat. Így bárki készíthet olyan titkos üzenetet, amit még ő maga sem tud kibontani – én viszont igen.
A digitális aláírás alapelve ezt fordítva használja: a titkosításhoz kell a két titkos prímszám, de a kibontáshoz már elég a szorzatuk. Ekkor csak az aláíró tudja előállítani a titkos üzenetet (aláírást), de bárki kibonthatja. Ez. Amikor digitálisan aláírok valamit, akkor a csak általam ismert, de hitelesített forrásból származó kulccsal titkosítom a saját adataimat, a dátumot az aláírandó dokumentum adatait, esetleg a dokumentum tartalmából készített valamilyen „vízjelet”, és még más azonosítókat. Az így készült titkos adatcsomagot pedig csatolom a dokumentumhoz, így az alá van írva.

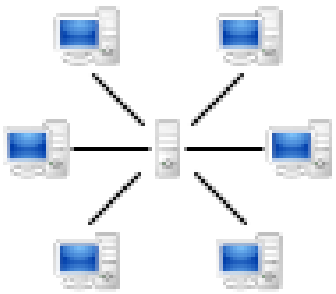
Távoli elérés

Programok futtatása egy másik számítógépről hálózaton keresztül. Erre eredetileg a **Telnet** protokoll szolgált, amely egy karakteres üzemmódú képernyős terminált igyekszik utánozni. Biztonságtechnikai problémái miatt (jelszavakat kódolatlanul küldi át a hálózaton) manapság egyre inkább kiszorítja az **SSH** (Secure Shell), amely a Telnet funkcióit erős titkosítási és felhasználó-azonosítási képességekkel egészíti ki.
A rendszert továbbfejlesztették, és ma már az **SSH segítségével grafikus felület is átvihető.**
A Windows rendszeren belül saját protokoll használható, az úgynevezett „Távoli asztal elérés”, amikor egy másik PC képernyője jeleníthető meg és kezelhető a saját képernyőn.

File csere - P2P (peer-to-peer)



P2P hálózat



Szerver-kliens hálózat

Elsősorban audio-vizuális anyagok terjesztésére alakították ki a fájl cserélő rendszereket. A legismertebb a Napster.
Sokáig a meghatározó struktúra a kliens-szerver elrendezés volt. Ilyenkor a szerver tárolja az anyagokat, a kliensek pedig a szerverről töltik le azokat. A megoldás egyszerű, de a szerver kapacitása limitálja a letöltési sebességet – így végső soron lassul, illetve betelik a rendszer.
A peer-to-peer (p2p) technika az információt elosztottá teszi, így mind a sávszélesség-, mind pedig a tárolókapacitás-problémát át tudja hidalni (peer somebody: egyenrangú valakivel).
Az eljárás lényege, hogy **az információ el van osztva, a közreműködő gépek mind egyenrangúak, és közvetlenül egymással kommunikálnak**. Így kevésbé alakul ki szűk keresztmetszet. Mindegyik gép megosztja a többivel a saját adatait, tehát **az egyes gépek tárolókapacitása összeadódik**.
A p2p rendszer legelső változata a Napster-hez fűződik. Ez a rendszer ugyan még használt központi szervereket a fájlok nyilvántartásához, valamint sok gyerekbetegséggel küszködött, mégis igen gyorsan sikeressé vált. A ma használatos p2p rendszerek (DirectConnect, KaZaA, BitTorrent, stb.) már meglehetősen optimálisan használják ki az erőforrásokat, és egyre inkább igazságosságra törekszenek.
Fontos megjegyezni két dolgot:

1. Az egyes p2p hálózatokhoz többféle kliensprogram is rendelkezésre áll.
2. A védett tartalmak megosztása ellentétes (lehet) a szerzői joggal.

A peer-to-peer hálózatok másik jelentős alkalmazási területe a telefonprogramok, például a Skype. Ezek a peer-to-peer hálózatok által olcsón biztosított infrastruktúrát használják ki, hogy olcsón szolgáltassanak telefonos kapcsolatot. A peer-to-peer hálózat miatt jelentősebb befektett tőke nélkül tudnak olyan szolgáltatást nyújtani, amelyek a korábban óriási pénzekből létrehozott infrastruktúra megtérülése miatt maradtak drágák.
A Skype készítői a telefonprogram fejlesztése során felismerték a peer-to-peer hálózatokban rejlő lehetőségeket, és 2006 decemberében kísérleti jelleggel beindították Joost néven az egyik internetes televíziós szolgáltatást.
Jelenleg kutatások és próbálkozások folynak a P2P megközelítés szélesebb körű alkalmazására. Például, a masszív többszereplős online játékok következő generációi valószínűleg erre a technológiára is alapoznak majd. Az egy realm-ban (azaz egy szerveren) együtt levő játékosok számát technikai korlátok szűkítik, például a szerverek sávszélessége, a szerverek száma, a hardver teljesítménye. A hagyományos egy szerver - sok kliens megközelítésben rendkívüli költségek lépnének fel a további játékosszám-bővítésre. A P2P megközelítéssel tervezett MMO játékok szinte korlátlan méretű realmok felépítésével kecsegtetnek.

Műsorszórás

Műsorszórásra a legtöbb esetben az UDP protokollt használják, mivel az gyors és hatékony, az elveszett/duplázódott csomagok pedig általában nem okoznak gondot.
Ma már sok rádióadó és TV adás fogható az Interneten, s a növekvő sávszélességnek köszönhetően ezek egyre élvezhetőbb formában jutnak el a hallgatókhoz, nézőkhöz. (A sávszélesség rohamosan nő! Megjegyezzük, hogy a normál minőségű TV kép átviteléhez 4-6 Mbit/s sávszélesség szükséges.)