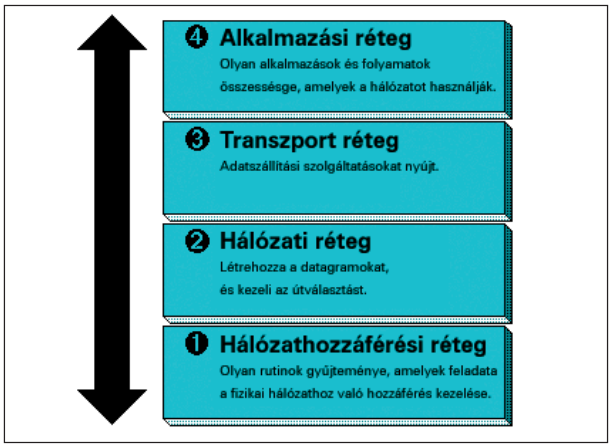
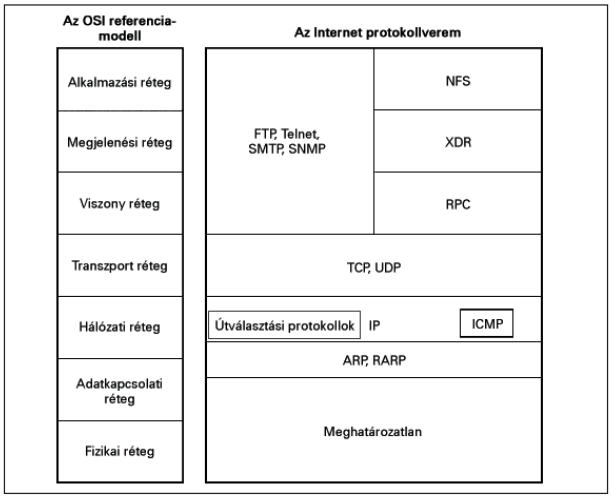
Számítógép-hálózatok (19. rész)
Az Internet hálózati rétege, az IP protokoll, címzések és alhálózatok

A sorozat elmúlt néhány részében áttekintettük, mi is a feladata, illetve miként is működik a hálózati réteg. Most egy konkrét megvalósítást veszünk szemügyre, amelyen megnézzük, miként is mennek a dolgok a gyakorlatban. Ebben a hónapban tehát témánk tehát az Internet hálózati rétege és annak protokolljai.

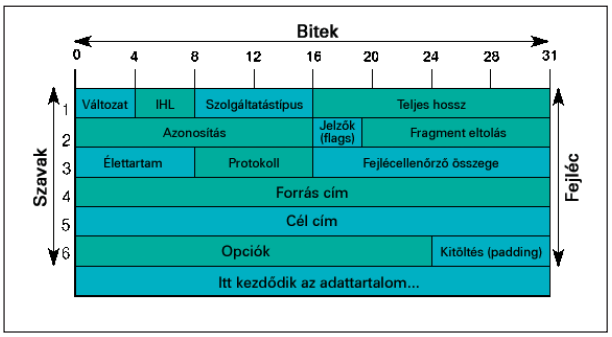
A hálózati réteg feladata nem más, mint a szállítási rétegtől kapott adatokat eljuttassa a címzettig. Mindezt úgy kell megvalósítania, hogy a szállítási réteg szempontjából teljesen mindegy legyen, hogy a cél a forrással egy hálózatban van-e, vagy sem. Magyarul a szállítási réteg csak az elküldendő adatot és a cél címét adja meg, minden más a hálózati rétegre van bízva.
Az Interneten zajló kommunikáció tehát a következőképp zajlik (*1. ábra*): a ***szállítási réteg (transport layer)*** kap egy adatfolyamot az ***alkalmazási rétegtől (application layer)***. Ezt az adatfolyamot blokkokra, úgynevezett datagramokra bontja, majd átadja a ***hálózati rétegnek (network layer)***, amely gondoskodik azok célbajuttatásáról. Látható, hogy az Internet felépítése nem követi az ***OSI*** modellt, amelyre sorozatunkat is építettük. A fizikai és az adatkapcsolati réteg ugyanis össze van vonva, továbbá hiányzik a viszont és a megjelenítési réteg (ezek egyes feladatai a szállítási, mások pedig az alkalmazási rétegben kerülnek megvalósításra).
A *2. ábrán* láthatjuk az Internet protokolljainak egymáshoz való viszonyát. Ezek közül a mezei felhasználók csak az alkalmazás rétegbeli protokollokkal kerülnek közelebbi kapcsolatba, mint például az ***FTP***-vel vagy az ábrán ugyan fel nem tüntetett, ám manapság leggyakrabban használt ***HTTP***-vel, amely a weboldalak lehívásában játszik szerepet. A ***TCP (Transmission Control Protocol)*** két gép közötti megbízható bájtfolyam alapú kommunikáció kialakítására alkalmas protokoll, az ***UDP (User Datagram Protocol)*** pedig ennek megbízhatatlan „párja”. Ezek a szállítási réteg protokolljai, a későbbiekben részletesen is foglalkozunk velük. Mi most azonban még maradunk a hálózati rétegnél, és azok protokolljainál. Közülük a legjelentősebb az ***IP (Internet Protocol)***, amely a gépek közötti adatátvitelt végzi. Fontosak még a vezérlőprotokollok, mint például az ***ICMP (Internet Control Message Protocol)***, vagy a logikailag az adatkapcsolati és a hálózati réteg határán elhelyezkedő ***ARP*** és ***RARP***.



1. ábra Az Internet rétegei. Ez a felépítés némiképp eltér az OSI modelltől, hiszen hiányzik az adatkapcsolati, a viszony és a megjelenési réteg.



2. ábra Az Internet protokolljai



3. ábra Az IP csomagok fejlécének felépítése

Az ennél mélyebb szinten lévő dolgokról az Internet szabványa már semmit sem mond. Itt az adott ***LAN***-ra van bízva, hogy miként valósítja meg a benne lévő gépek közötti adatátvitelt. Az ***Internetbe*** tehát ugyanúgy beköthetünk például egy vezérjeles gyűrűt, mint egy ***Ethernet*** hálózatot. Most pedig vegyük kicsit részletesebben is szemügyre az ***Internet*** hálózati rétegének protokolljait!

Az IP (Internet Protocol)

A szállítási réteg által előállított ***TCP*** és ***UDP*** datagramok ***IP*** csomagok belsejében utaznak a világhálón. Egy ***IP*** csomag két részből áll: magából a szállított adatból és egy fejlécből. A fejléc szerkezetét láthatjuk a *3. ábrán*.

Amikor két különböző számítógép, például egy ***SPARC*** és egy ***Pentium*** processzorral felszerelt masina bitsorozatokat szeretne egymással cserélni, akkor rögtön félreértések adódnak. A ***Pentium*** ugyanis alsó vég szerint tárolja a biteket, azaz mindig a legkisebb helyiértékű bitet küldi elsőnek. A ***SPARC***-nál ez pont fordítva történik. Ha a ***Pentium*** azt akarja mondani partnerének, hogy „41”, akkor ő az 100101 bitsorozatot fogja küldeni, amit a ***SPARC*** 37-nek fog értelmezni. Ezért először meg kell egyezni abban, hogy a biteket alsó- vagy felső vég szerint továbbítjuk. Az ***IP*** protokoll felső vég szerint tárolja az adatokat a fejlécében. (Ezért ha ***Pentium*** processzorra írunk hálózati alkalmazást, figyelnünk kell arra, hogy a cél címét először átalakítsuk felsővégűre, majd csak utána adjuk át a szállítási rétegnek).

Most nézzük sorra, mi mit is jelent a fejlécben. A verzió értelemszerűen az ***IP*** protokoll verzióját adja meg, pontosabban azt, hogy az adott csomag az ***IP*** protokoll mely változatához tartozik. Az ***IHL*** mező a fejléc utolsó részének, az opciók méretét adja meg. Erre azért van szükség, mert ennek a mezőnek nincs kötött mérete, hossza csomagonként változó lehet. Mindenesetre 60 bájtnál nagyobb nem lehet.

A ***szolgáltatás típusa (type of service)*** eredetileg arra hivatott, hogy az alhálózat számára hordozzon információkat a továbbításánál felmerülő kérdésekkel kapcsolatban. Ilyen lehet például a csomag prioritása (erre az ***IP*** 8 szintet biztosít), vagy például az, hogy gyorsan, vagy inkább biztonságosan kívánjuk a csomagot céljához eljuttatni. Példaként most is felhozhatjuk az előző részben leírtakat: egy hálózaton keresztüli videónézésnél fontosabb, hogy a csomagok sebessége ne lassuljon le, ugyanakkor egy fájl átvitelekor inkább az a jobb, ha a csomagok sértetlenül érkeznek meg. Az igazsághoz azonban az is hozzátartozik, hogy az útválasztók többsége nem foglalkozik az itt megadott paraméterekkel. Ez gyakorlatilag azt jelenti, hogy bármit is írunk erre a mezőre, nem lesz kihatással a csomagunk célbajutásának módjára.

A ***teljes hossz (total length)*** az egész csomag méretét adja meg, beleértve a fejlécet és a szállított adatokat is. Mivel ez egy két bájtos mező, ebből következően egy csomag nem haladhatja meg a 65535 bájtot (egy bájt híján 64 kilobájt). Ez első hallásra soknak tűnhet, de a sávszélesség növekedésével előbb-utóbb kevésnek számítanak majd, és nem lesz gazdaságos ilyen „kis” csomagmérettel dolgozni.

Az útválasztók időnként feldarabolják a datagramokat. Az ***azonosítás (identification)*** mező adja meg, hogy az adott csomag melyik datagram része. A ***jelzők (flags)*** három bitből áll, amelyek közül az első kihasználatlan, a második az úgynevezett ***DF (Don’t Fragment – ne darabold!)*** bit. Ez megtiltja az útválasztók számára, hogy az adott csomagot több részre szabdalják. Ezt a bitet akkor szokás beállítani, amikor a célállomás valami oknál fogva nem képes több darabból összeállítani az eredeti datagramot. A darabolás megtiltásának azonban ára van: elképzelhető, hogy a csomag egy darabban csak kerülőúton keresztül érhet célba. A jelzők mező harmadik bitje az ***MF (More Fragments – még több darab)***. Ez azt jelzi a célállomás számára, hogy a datagram még nem ért át teljesen, újabb darabokra kell számítani.

Ahhoz, hogy a gép rekonstruálhassa a számára küldött datagramot, a darabok sorrendjét is ismernie kell (hiszen semmi sem garantálja, hogy a csomagok útnak indulásuk sorrendjében érkeznek meg). Erre szolgál a ***darabeltolás (fragment offset)*** mező, amely megmondja, az adott darab mely részét képezi a datagramnak. Az utolsó darabot leszámítva ennek a mezőnek az értéke mindig a 8 valamely egész számú többszöröse. Mivel az egész mező összesen 13 bájt méretű, ezért egy datagramot 8192 darabnál többre nem oszthatunk.

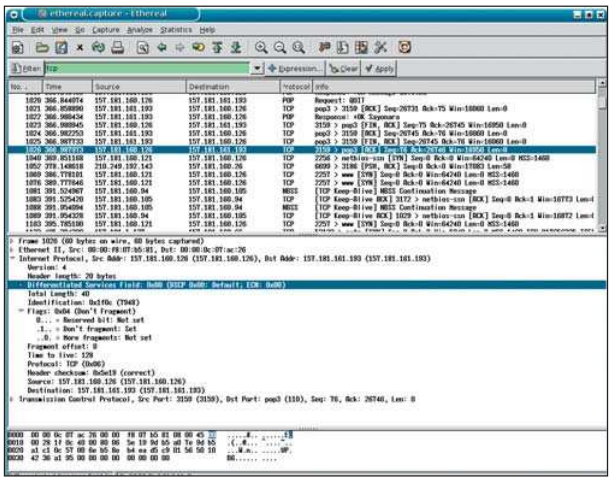
Az alhálózaton örökre bolyongásra ítélt csomagok elkerülésére szolgál az ***élettartam (time to live)*** mező, amely megmondja, még mennyi ideig élhet az adott csomag. Ha ez az érték nullára csökken, az útválasztók a csomagot eldobják. A szabvány szerint az élettartamot másodpercben kell megadni, de a gyakorlatban inkább ugrás-számban határozzák meg, azaz minden útbaeső útválasztó eggyel csökkenti a mező értékét. Ha egy csomag ideje lejár, az útválasztó, amelyik a csomagok eldobta, egy értesítőt küld a feladónak a történtekről.

A protokoll rész arról árulkodik, hogy a csomag milyen típusú datagramot hordoz, például ***TCP***, ***UDP***, stb. Ez a célállomás számára fontos, hiszen annak hálózati rétege innen tudja, hogy mely szállítási réteghez tartozó folyamatnak kell a beérkező adatokat átadnia.

A ***fejlécellenőrző összeg (header checksum)*** értelemszerűen a fejléc sértetlenségének megállapítására szolgál. Mivel a fejléc élettartam mezője minden ugrásnál változik, ezért az útválasztóknak a csomag továbbítása előtt az ellenőrző összeges újra ki kell számolniuk.

A ***forrás és a cél címe (source/destination address)*** megmondja, ki a címzett és ki a feladó. Az ***Interneten*** a gépek úgynevezett ***IP*** címekkel vannak azonosítva, erre mindjárt részletesen is kitérünk.

4. ábra A traceroute parancs segítségével feltérképezhetjük, hogy milyen útválasztókon keresztül érhetjük el a megadott célállomást



5. ábra Egy csomag belső felépítése

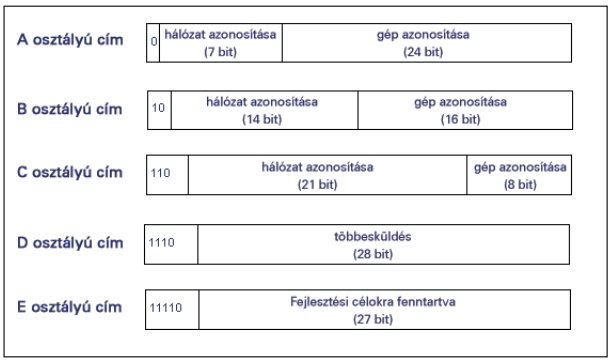
Az ***opciók (options)*** rész eredetileg arra szolgált, hogy később jelentősebb változtatások nélkül bővíthessék az *IP* protokoll által nyújtott szolgáltatásokat. Mint már említettük, ez a rész változó méretű, lehet teljesen üres is (de 40 bájtál semmiképp sem több), így csak akkor kell több helyet lefoglalni a fejrész számára, ha az opció mező valóban felhasználásra is kerül.
Az *IP* tervezése óta öt dologgal bővítették ki az *IP* fejlécét, illetve pontosabban az opció mezőt. Ezek közül az első a biztonság, amellyel meghatározhatjuk a csomagban szállított datagram titkosságát. Az eredeti célkitűzés az volt, hogy bizonyos titkosnak számító adatokat csak meghatározott útválasztókon keresztül lehessen továbbítani. Ezzel elkerülhető például az, hogy katonai titkokat hordozó csomagjaink áthaladjanak ellenséges (vagy legalábbis nem túl barátságos) országok útválasztóin. A ma használatban lévő útválasztók nem támogatják ezt a szolgáltatást.
Egy másik opció a szigorú forrás általi forgalomirányítás, amikor a datagramot küldő gép határozza meg, hogy a csomagnak milyen útvonalon kell eljutnia a célállomásáig. Ilyenkor az opciórészben meg kell adni a pontos útvonalat (útválasztók IP címeinek sorozatát), amelyen a csomagnak haladnia kell. Ennek egy „enyhébb" változata a laza forrás általi forgalomirányítás, amikor csak azoknak az útválasztóknak a címeit mondjuk meg, amelyeken a csomagnak biztosan át kell haladnia. Ezzel az opcióval lehetőség nyílik arra, hogy csomagjainkkal elkerüljünk bizonyos térségeket.
A negyedik opció az útvonal feljegyzése. Ennek hatására az útválasztók az átmenő csomagok opció mezejébe beírják saját címüket, így a célállomás rekonstruálhatja, milyen útvonalon érkeztek meg hozzá a csomagok. A ***traceroute*** parancs (*4. ábra*) is hasonló szolgáltatást nyújt, segítségével kilistázhatjuk, milyen útválasztókon kell keresztülmennünk ahhoz, hogy elérjünk egy megadott gépet. A különbség csak az, hogy a ***traceroute*** nem használja az útvonal feljegyzése opciót, teljesen más elven működik. Az az igazság, hogy nem is lehet ezt a módszert erre a célra használni, ugyanis ezt még akkor találták ki, amikor az Internet mérete még jóval kisebb volt, és 9 ugráson belül bárhova el lehetett jutni. Ma már más a helyzet. A 4. ábrán látható, hogy például egy népszerű amerikai keresőportál eléréséhez a csomagoknak legalább 17 útválasztón kell keresztülverekedniük magukat. Ilyen hosszú útvonal pedig már aligha fér el 40 bájton.
A ***traceroute*** inkább azt használja ki, hogy az útválasztók visszajeleznek, ha egy általunk küldött csomagnak lejár az életideje, és el kell dobniuk. Így kiküld különböző ám kis élettartamú csomagokat (amelyek annyi idő alatt biztosan nem jutnak el céljukig), és amelyik útválasztó visszajelzi a csomag halálhírét, az biztos része a csomag útvonalának.
Végezetül megemlítjük még az utolsó opciót is, az időbélyeget, amely szinte teljesen megegyezik az előzővel, csak azzal a különbséggel, hogy az útválasztók nem csak az *IP* címüket, hanem a csomag beérkezésének idejét is feljegyzi. Ezáltal pontosabb képet kaphatunk a csomagok alhálózatbeli terjedéséről.

Az ICMP (Internet Control Message Protocol)

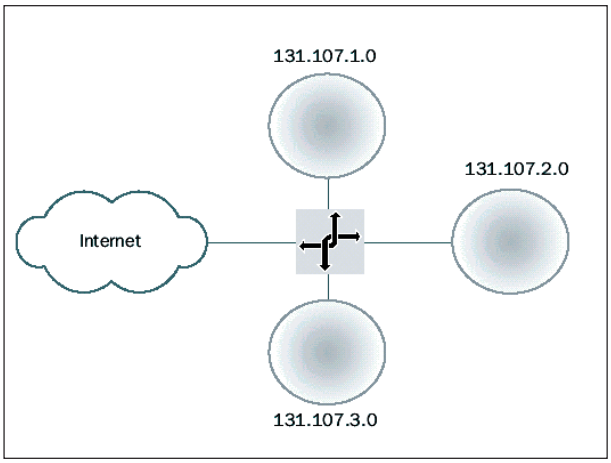
Az *IP*-n kívül más protokollokat is használ az ***Internet*** hálózati rétege. Ezek közül az egyik legfontosabb az *ICMP*, amelynek két funkciója van. Először is lehetővé teszi hogy mérhessük a hálózat bizonyos tulajdonságait. Másodszor lehetőséget biztosít az útválasztók számára, hogy jelezzék a csomag feladójának, ha valami váratlan esemény következik be.
Ilyen váratlan esemény lehet például az, ha az útválasztó nem találja magát a célt, vagy hibás fejlécű *IP* csomagot kapott. Az *ICMP* üzenetek másik gyakori felhasználása az, amikor egy gép életjeleit kívánjuk megállapítani (például a ***ping*** parancs segítségével). Ilyenkor egy úgynevezett *ECHO_REQUEST* üzenetet küldünk, amelyre a gép – amennyiben életben van – egy *ECHO_REPLY ICMP* üzenettel válaszol.

Címzés az Interneten

Ahhoz, hogy a hálózati réteg megtalálja a csomagok célállomását, minden gépnek (és útválasztók) rendelkeznie kell egy vagy több egyedi azonosítóval. Az ***Interneten*** ez az azonosító az *IP* cím, amely egy 32 bites összeg.



6. ábra Az IP címosztályok



7. ábra Alhálózatok egy lehetséges felosztása

Ezt a számot általában bájtonként decimálisan, pontonként elválasztva szokás megadni, például: 195.38.96.96. A valamivel több mint 4 milliárd lehetséges ***IP*** címet öt csoportra oszthatjuk (*6. ábra*). Ezek közül az első három osztály (A, B és C) ***IP*** címei azonosítanak gépeket. Az egy hálózatban lévő hosztok hálózati címeinek tehát meg kell egyezniük. A D osztályba tartozó címek a ***többesküldésre (multicasting)***, az E-beli címek pedig későbbi felhasználásra vannak fenntartva.

Hogy egy ***IP*** cím melyik osztályba tartozik, azt az első pár bitje adja meg. Az A osztályú címek például mindig 0-val kezdődnek, ez azt jelenti, hogy az 1.0.0.0 – 127.255.255.255 tartományon belül minden ***IP*** cím A típusú. Az A, B és C osztályok közötti különbség az, hogy mennyi bitet használnak magának a hálózat, és mennyi bitet a hálózatban lévő egyes gépek azonosítására. Az A osztály esetében összesen 126 hálózatot lehet megkülönböztetni, viszont ezek a hálózatok akár 16 millió gépből is állhatnak. (Persze ez azt is jelenti, hogy összesen csak 126 darab A osztályú címcsoport osztható ki). Ezzel szemben a C típusú címek már 21 bitet használnak a hálózaton azonosítására, viszont csak 256 (gyakorlatban 254, lásd később) darab gépet tartalmazhatnak.

Vannak speciális célra fenntartott ***IP*** címek is, amelyek nem oszthatók ki a gépek számára. Ilyen például a csupa egyes bitből álló (255.255.255.255), amely az adatszórást teszi lehetővé a helyi hálózaton. Azok az ***IP*** címek, amelyeknek a hálózati része csupa nullából áll, mindig az aktuális hálózatra vonatkoznak. A 127.*.*.* tartomány szintén nem osztható ki, ez az úgynevezett ***visszacsatolás (loopback)***, amelyeken keresztül a gép saját magát „érheti el”. Próbáljuk csak megpingelni bármelyik 127-el kezdődő ***IP*** címet, mindig a saját gépünk fog válaszolni. A visszacsatolásnak például akkor vehetjük hasznát, ha ki szeretnénk próbálni egy hálózati alkalmazást anélkül, hogy egy második gépet is igénybe vennénk.

Alhálózatok

Már említettük, hogy az egy hálózatba tartozó gépek hálózati címeiknek meg kell egyezniük. Egy B osztályú címtartományt birtokló szervezet esetében azonban nincs mind a 65 ezer gép egy fizikai hálózaton. Egy egyetem esetében például külön hálózatot képezhetnek az egyes géptermek, illetve a tanszékek számítógépei, és ezek a különálló ***LAN***-ok hidak segítségével vannak összekapcsolva. Felmerül a kérdés, hogy akkor miként osszuk ki az ***IP*** címeket?

Kézenfekvő megoldás lehet az, hogy visszaadjuk a B címtartományunkat, és kérünk annyi C osztályút, ahány hálózatunk van. Ezzel azonban több problémánk adódna, például jelentősen megnehezülne a hálózat menedzselése. A megoldás az ***alhálózatok (subnets)*** létrehozásában rejlik. (Az elnevezés sajnos megtévesztő. Ezeknek az alhálózatoknak semmi köze az útválasztókból álló kommunikációs alhálózathoz).

Az alhálózatok kialakításához az ***IP*** címek gépek azonosítására szolgáló 16 bitjéből leválasztunk valamennyit, amelyek az egyes alhálózatokat fogják megadni. A *7. ábrán* láthatunk is erre egy példát. Tegyük fel, hogy a 131.107.*.* B típusú ***IP*** címtartománnyal rendelkezünk. Itt az utolsó két bájt azonosítja a gépeket. Az ábrán ebből 8 bitet az egyes alhálózatok azonosítására áldozunk. Ekkor összesen 254 alhálózatunk lehet, és mindegyikben 254 gép helyezhető el.

Fontos, hogy ez a felosztás kívülről (egy másik hálózatból) nem látszik. Az ottani szemlélődő úgy látja, hogy nekünk csak egyetlen nagy hálózatunk van. A belső útválasztóink azonban ismerik a hálózatunk belső felépítését, és tudják, melyik gép merre található. Pontosabban csak azt kell tudniuk, hogy az egyes alhálózatok merre találhatóak. Az úgynevezett alhálózati maszk segítségével ugyanis az útválasztók az ***IP*** címből ki tudják számolni, hogy melyik alhálózat felé is kell terelni a csomagot. Az alhálózati maszk egy olyan 32 bites szám, amellyel ha össze ÉS-eljük az ***IP*** címmel, akkor kinullázódik annak a gépet azonosító része.

Az előző példánknál a ***netmask*** minden bizonnyal a 27 darab egyesből, és 8 darab nullából (255.255.255.0) lenne. Ha beérkezik egy csomag, amely a 131.107.12.2-es című gépnek szól, akkor az útválasztó az alhálózati maszkkal „összeéselve” megkapja, hogy a kérdéses gép melyik alhálózatban is található (jelen esetben a végeredmény a 131.107.12.0 lesz).

A következő részben megismerkedünk az ***Internet*** más fontos, szintén a hálózati rétegbe tartozó protokolljaival, például az ARP-al és a RARP-al.

Garzó András
garzoand@interware.hu