# Számítógépes hálózatok – 5. előadás

## Adatkapcsolati (link) réteg, folytatás

### PPP - Point to Point Protocol

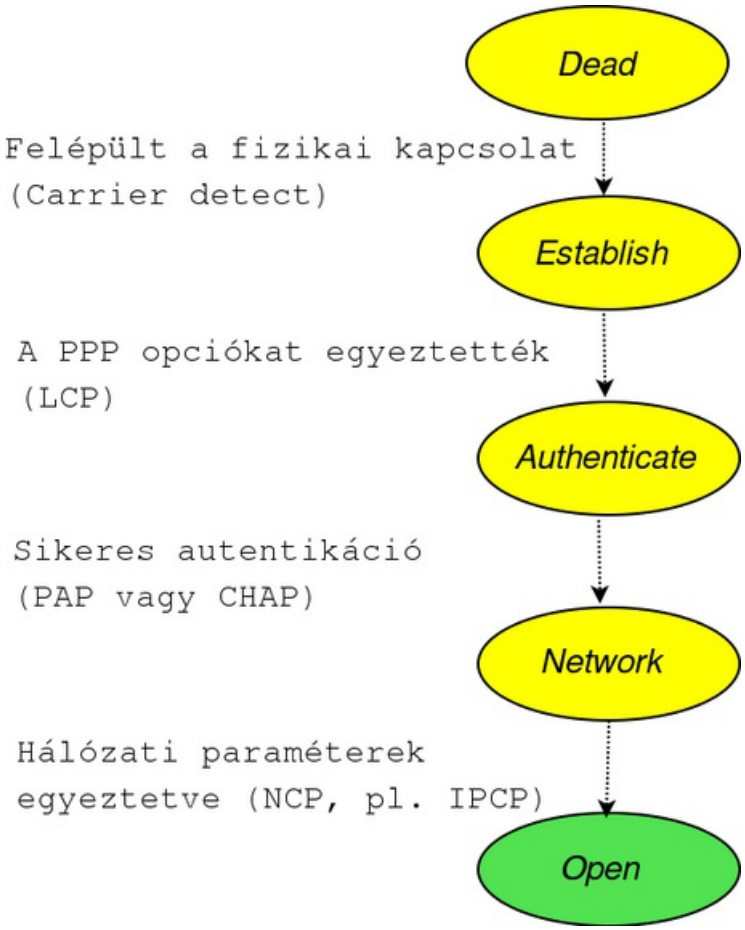
- **RFC1661**: pont-pont kapcsolatra használatos adatkapcsolati réteg
- IP alatt gyakran használt, másra is használható
  - Az első hasonló protokoll a SLIP volt (Serial Line IP, **RFC1055**)
- Klasszikus esetben telefon vonalon
- Feladatok — ezeket tűzték ki, amikor tervezték a protokollt (**RFC1547**)
  - Frame-ek elkülönítése
  - Transzparencia: tetszőleges byte folyamot át lehessen vinni
  - Hatékony sávszélesség kihasználás
  - Hiba felderítés (CRC)
  - Többfajta protokollt támogasson (ne csak IP-t), ezeket multiplexálni lehessen
  - Magasabb protokoll paraméterek egyeztetését oldja meg (pl. IP címek)
  - Többféle fizikai médiumon működhessen

### PPP frame szerkezet - ISO HDLC (High level Data Link Control) alapú

| Flag (1-byte) | Address (1-byte) | Control (1-byte) | Protokoll (2-byte) | Adat | CRC (2 byte) | Flag (1-byte) |
|---|---|---|---|---|---|---|

- Flag: a frame kezdetét és végét jelzi: `0x7E`
- Address: PPP-nél mindig `0xFF`
- Control: PPP-nél mindig `0x03`
- Protocol: ethernet „type”-hoz hasonló
  - LCP - Link Control Protocol
  - PAP, CHAP - Password Authentication Protocol, Challange Handshake Authentication Protocol
  - NCP - Network Control Protocol, pl. IPCP
  - Adat - pl. IP
- Adat
  - Általában legfeljebb 1500 byte. LCP-vel másban is megállapodhatnak
  - Az adatfolyamban a flag byte-ot escape-elni kell: byte stuff
    - Az escape byte: `0x7D`
    - Az escape-elt byte-ot XOR-oljuk `0x20`-szal
    - `7E → 7D 5E`
  - Alapértelmezésben minden `0x20`-nál kisebb byte-ot is escape-elnek
    - Pl. Ctrl/A: `01 → 7D 21`
  - Az escape byte-ot is escapeljük: `7D → 7D 5D`

### PPP állapotok




## LCP: Link Control Protocol

- Kapcsolat felépítés
- Kapcsolat bontás
- Paraméterek egyeztetése
  - Autentikálás
  - Tömörítés
  - Sallangok elhagyása
- Kapcsolat figyelés

## PAP: Password Authentication Protocol

- Két lépés
  - Kliens küldi az azonosítót és a jelszót (kódolatlanul)
  - Szerver válaszol, hogy rendben van-e
- Nem biztonságos
  - Lehallgatható
  - Visszajátszható

## CHAP: Challange Handshake Authentication Protocol

- **RFC1994**
- Három lépés
- A szerver egy véletlen számot küld a kliensnek (challenge)

- A kliens a kapott challenge-ből, és a jelszóból választ képez
- A szerver ellenőrzi
- Nem megy cleartext-ben a jelszó, nem visszajátszható a folyamat

## NCP: Network Control Protocol

- A magasabb réteg konfigurálására szolgál, több van (pl. külön Novell)
  - Egy időben akár több magasabb szintű kapcsolat lehet egy PPP session fölött
- IP: IPCP, IP Control Protocol, **RFC1332**
  - IP cím
  - Van Jacobson TCP/IP fejrész tömörítés
    - A rendszerint 40 byte-os TCP/IP fejrészből 3 byte lehet
  - DNS szerverek címe (**RFC1877**)

## PPPoE

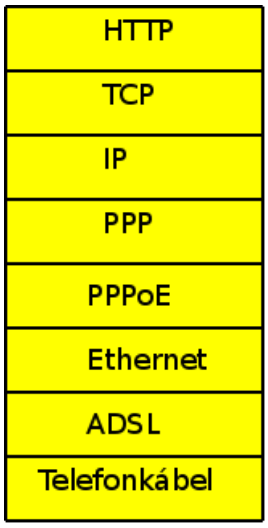
| Ver. (4 bit) | Típus (4 bit) | Kód (1 byte) | Session ID (2 byte) | Hossz (2 byte) | Adat |
|---|---|---|---|---|---|

- PPP over Ethernet, **RFC2516**
- ADSL és FTTH (Fiber To The Home) előfizetőknél ez használatos
- Ethernet type mező:
  - `0x8863`: discovery stage
  - `0x8864`: PPP session stage
- Discovery
  - A partnerek megtudják az ethernet címeket és egy session ID-t
  - Négy lépcsős folyamat (PAD = PPPoE Active Discovery)
    - Broadcast ethernet csomaggal indul **(PADI, Initiation)**
    - Több állomás (Access Concentrator) is válaszolhat **(PADO, Offer)**
    - A kliens egy unicast csomaggal dönt **(PADR, Request)**
    - A koncentrátor egy session id-t küld **(PADS, Session-confirmation)**

```
Kliens                              Koncentrátor

                PADI                                Ki segít
          ----------------->
                PADO                                Ha akaroc
          <-----------------

                PADR                                Kérlek se
          ----------------->

                PADS                                Rendben!
          <-----------------
```

- Session
  - Unicast csomagok, mindben ott a session id, hossz
  - PADT (Terminate) lezárhatja

## Több mint 7 réteg:

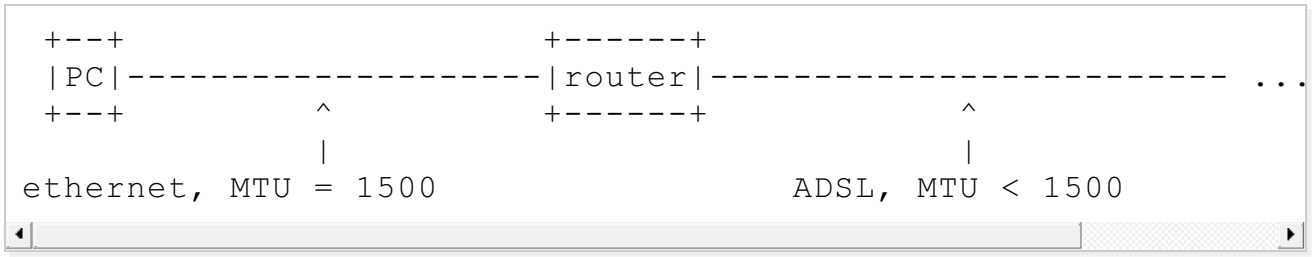



## Loopback interfész

- Sokszor a kliens és a szerver ugyanazon az eszközön van
- Akkor is akarunk hálózatot látni, ha nincs fizikai összeköttetés
- IP címe: `127.0.0.1`
  - Ez az IP cím fizikai hálózaton sose jelenhet meg
- Minden tekintetben úgy viselkedik, mint egy valódi interfész
- Az egyes szolgáltatásoknál rendelkezhetünk róla, hogy ezen az interfészen is szolgáltassanak-e
- Nem csak számítógépeknél, hanem más dobozoknál: switch-ek, routerek stb.

## MTU - Maximum Transmission Unit

- A fizikai médium rendszerint korlátozza az átvihető frame méretét
- Mértékegység: byte
- Célszerű ennél nem nagyobb csomagokat használni a felső rétegen
  - Nem tudhatjuk, hogy a célállomásig milyen közegen, milyen MTU-val megy át a csomag
  - Elvben lehet a legkülönbözőbb
  - Van TCP/IP implementáció, ami nem tolerálja :-(



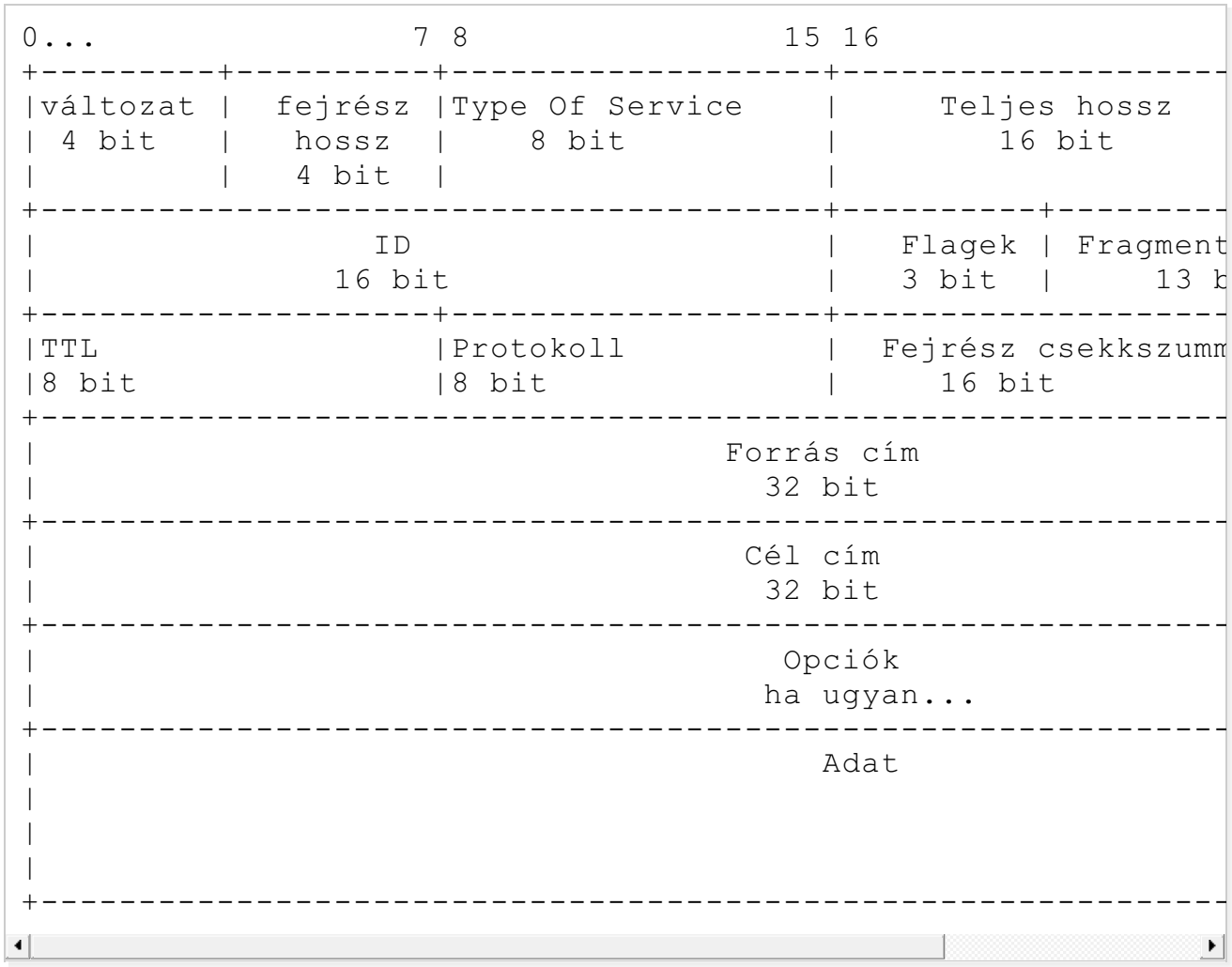

- Ethernet: 1500 a szokásos
- 802.3: 1492
- Path MTU: egy adatkapcsolatnál a közbülső legkisebb MTU

# IP, Internet Protokoll

- **RFC791** Postel, 1981

## Tulajdonságok

- Nem kapcsolatorientált
  - Az egyes csomagokat a hálózati réteg egymástól függetlenül kezeli
  - Nincs kapcsolat felépítés, kapcsolat bontás
- Best effort, decentralizált, nem megbízható
  - Nem biztos, hogy a célba ér a csomag
  - Nem biztos, hogy nem érkezik többször
  - Nem biztos, hogy egymás utáni csomagok sorrendje megmarad
  - Nem biztos, hogy nem sérülnek benne bitek
- A felsőbb rétegek gondoskod(hat)nak mebízható, kapcsolatorientált kommunikációról




- A 0. bit a legnagyobb helyiértékű (big endian)
- Az egyes sorokban álló byte-ok balról jobbra mennek át (big endian)
- A változat (version) klasszikus esetben 4, manapság lehet 6

## Fejrész hossz

- 4 bit, a mértékegység 4 byte.

## TOS

- **RFC791**

```
0     1     2     3     4     5     6     7
+-----+-----+-----+-----+-----+-----+-----+-----+
|   PRECEDENCE    |        TOS        |  0  |  0  |  RFC 791
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- **RFC1122**

```
0     1     2     3     4     5     6     7
+-----+-----+-----+-----+-----+-----+-----+-----+
|   PRECEDENCE    |        TOS                |  RFC 1122
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- **RFC1349**

```
0     1     2     3     4     5     6     7
+-----+-----+-----+-----+-----+-----+-----+-----+
|   PRECEDENCE    |        TOS              | MBZ |  RFC 1349
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- MBZ: Must Be Zero
- **RFC2474**

```
+-----+-----+-----+-----+-----+-----+-----+-----+
|          DS FIELD, DSCP           |     CU    | RFC 2474
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- CU: Currently Unused
- **RFC3168**

```
+-----+-----+-----+-----+-----+-----+-----+-----+
|          DS FIELD, DSCP           | ECN FIELD | RFC 3168
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- DSCP: differentiated services codepoint
- ECN: Explicit Congestion Notification
- A csomag hálózat belsejében való továbbítását befolyásoló mező
- Sokszor átdolgozták
- Nincs garancia arra, hogy a hálózati partnerek kezelik
  - Többnyire nincs baj abból, ha nem
- Hálózati átjárók, tűzfalak. routerek manipulálhatják

## Hossz

- Az egész IP csomag hossza byte-ban
- Max 65535
- Egy interfészen az MTU korlátozza
- Közbülső eszközök fragmentálhatnak
- Ethernetnél mutathat kevesebbet mint az ethernet csomag
  - Az ethernet csomag legalább 46 byte

## ID

- A csomagra jellemző egyedi szám
- Az internet ősi korszakában az implementációkban eggyel több mint a előző
  - Megjósolható -> visszaélésre ad módot
  - Manapság véletlen szám
- Tűzfalak tovább randomizálhatják

## Flag-ek

- Az IP darabolásnál (fragmentálás) van szerepük
  - Don't Fragment (DF)
  - More Fragments (MF)

## Fragmentum offszet

- Ha fragmentált az IP datagram, azt mutatja, hogy ez a darab hova illik
- Mértékegység: 8 byte

## TTL - Time To Live

- Az időt hop-ban méri
- Felső korlátot ad közbülső routerekre
- Minden router dekrementálja
- Ha 0, eldobja a csomagot, ICMP üzenetet küld vissza a feladónak
- Ha nem lenne, végtelen ideig keringhetnének csomagok
- A traceroute program épp ezt használja
  - Egy sikertelen traceroute: **traceroute-fennakad.cap**
  - Egy sikeres (tcptraceroute): **tcptraceroute.cap**

## Protocol

- Az IP feletti protokollt adja meg
- IANA (Internet Assigned Numbers Authority) osztja
  - ICMP -1
  - TCP - 6
  - UDP - 17
  - Összesen: **http://www.iana.org/assignments/protocol-numbers/**

## Csekszumma

- Nem ethernet CRC, sima bit összeadás
  - **RFC1071**, **RFC1624** kiegészíti
  - ones' complement összeadás, a csekkszumma az eredmény egyes komplemense
  - Az ellenőrző oldalon 0 jön ki, ha minden jó
- Csak a fejrészre
- A csomag többi részét a felsőbb szint ellenőrizheti
- Hop-ról hop-ra változik. Miért?

## Forrás és célcím

- Az IP réteg szempontjából a legfontosabb: ennek alapján továbbítja (route-olja) a csomagot
- Tűzfalak még ezt is módosíthatják
- Pontozott decimális jelölés
- A négy byte-ot decimális számként írjuk, közéjük ponttal, pl.: 193.239.149.4

## Opciók


- Elvben több is lehet
- A mindennapi gyakorlatban ritka
- Pl: source routing
  - Manapság tiltják a routerek, tűzfalak

Szerző: Pásztor Miklós, Máray Tamás