

## Számítógépes Hálózatok 2010

### 13. Biztonság – tűzfalak; Felhasználói réteg – DNS, email, http

## Elektronikus aláírás

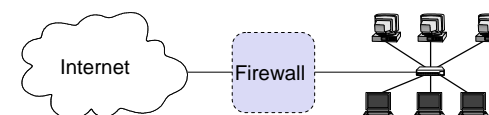
- Más néven digitális szignatúra
  - Az aláírónak van egy (titkos) privát kulcsa
  - A dokumentumot a privát kulcs felhasználásával írja alá
  - és a nyilvános kulccsal verifikálható, hogy az aláírás tőle származik
  - A nyilvános kulcs mindenki számára ismert
- Példa egy aláírás sémára
  - `text`: üzenet
  - Az aláíró
    - kiszámítja  $h(\text{text})$  értékét egy  $h$  kriptografikus hash függvénnyel
    - nyilvánosságra hozza `text` és `signature = g(\text{privat}, h(\text{text})) értékét, ahol  $g$  az asszimmetrikus visszakódoló függvény`
  - Az aláírást ellenőrző
    - kiszámítja  $h(\text{text})$  értékét
    - és megvizsgálja, hogy  $f(\text{public}, \text{signature}) = h(\text{text})$ , ahol  $f$  az asszimmetrikus titkosító függvény

## IPsec (RFC 2401)

- Védelem Replay-támadással szemben
- IKE (Internet Key Exchange) Protokoll
  - Megegyezés egy Security Association-ról (SA)
    - Identifikáció, rögzítése a kulcsoknak, hálózatoknak, az autentifikálás és az IPsec kulcs megújítási időközeinek
  - Egy SA létrehozása gyors üzemmódban (a megalapítása után)
- Encapsulating Security Payload (ESP)
  - IP-fejléc titkosítás nélkül, adatok titkosítva, autentifikálással
- IPsec szállítói módban (direkt kapcsolatokhoz)
  - IPsec-fejléc az IP-fejléc és az adatok között
  - Vizsgálat az IP-Routerekben (azokban IPsec-nek jelen kell lenni)
- IPsec alagút (tunel) módban (ha legalább egy router IPsec nélkül közben van)
  - Az egész IP csomag titkosított és az IPsec-fejléccel együtt egy új IP csomagba pakoljuk
  - Csak a végpontokban kell lenni IPsec-nek.
- IPsec része IPv6
- IPv4-portolás létezik

## Internet tűzfalak (firewalls)

- Egy hálózati tűzfal
  - Korlátozza a belépést a hálózatba egy gondosan ellenőrzött pontra
  - Véd, hogy a támadók ne jussanak más védelmi mechanizmusok közelébe
  - Korlátozza a kilépést egy gondosan ellenőrzött pontra
- Általában egy hálózati tűzfal egy olyan pontra van telepítve ahol a védett (al)hálózat egy kevésbé megbízható hálózathoz kapcsolódik
  - Pl.: egy belső „corporate local area network” és az Internet



- Aljában, tűzfalak hozzáférés ellenőrzést realizálnak a (al)hálózathoz

## Tűzfalak -- típusok

- Tűzfalak típusai
  - Host-Firewall
  - Hálózat-Firewall
- Hálózat-Firewall
  - megkülönböztet
    - Külső hálózatot (Internet: ellenséges)
    - Belső hálózatot (LAN: megbízható)
    - Demilitarizált zónát (a külső hálózatról elérhető szerverek)
- Host-Firewall
  - pl. Personal Firewall
  - Felügyeli a számítógép teljes adatforgalmát
  - Védelem külső és belső támadásoktól (pl. trojai)

## Tűzfalak -- módszerek

- Módszerek
  - Csomagszűrő (packet filter)
    - Portok vagy IP címek letiltása
  - Tartalomszűrő (content filter)
    - SPAM-Mailek, Vírusok kiszűrése, vagy ActiveX vagy JavaScript kiszűrése a HTML oldalakból
  - Proxy
    - Transzparens (kívülről látható) Host-ok
    - A kommunikáció és a lehetséges támadások elvezetése biztosított számítógépekre
  - NAT, PAT
    - Network Address Translation
  - Bástya Host

## Tűzfalak -- fogalmak

- (Network) Firewall
  - A hozzáférést az Internetről egy biztosított hálózatra korlátozza
- Csomagszűrő (packet filter)
  - Csomagokat választ ki a hálózatba menő vagy a hálózatról érkező adatfolyamból
  - Erkező csomagok szűrésének célja:
    - pl. a hozzáférés kontrolljának megsértésének felismerése
  - Kimenő csomagok szűrésének célja:
    - pl. Trojai felismerése
- Bástya host
  - Egy olyan számítógép a periférián, ami különös veszélynek van kitéve
  - és ezért különösen védett
- Dual-homed host
  - Közöséges számítógép két interfésszel (összeköt két hálózatot)

## Tűzfalak -- fogalmak

- Proxy (helyettes)
  - Speciális számítógép, amelyen a kérések és válaszok keresztül vannak irányítva
  - Előny
    - Csak ott kell védelmet biztosítani
- Network Address Translation (NAT):
  - lásd a következő fóliát
- Perimeter Network:
  - Egy részhálózat, amely a védett és védetlen zóna között egy további védelmi réteget ad
  - Szinoníma: demilitarizált zóna (DMZ)

## NAT és PAT

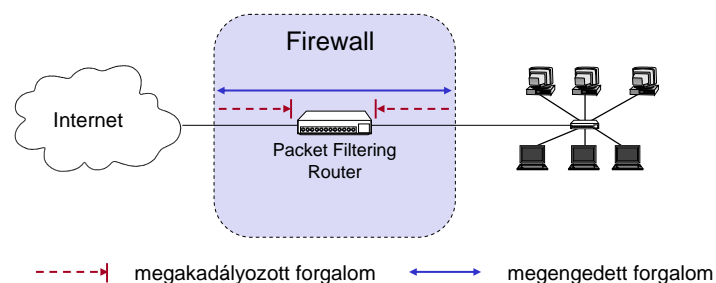
- NAT (Network Address Translation)
- Basic NAT (Static NAT)
  - Minden belső IP cím egy külsővel helyettesítődik
- Hiding NAT = PAT (Port Address Translation) = NAPT (Network Address Port Translation)
  - A socket-pár (IP-cím és Port-szám) átszámítódik
- Módszerek
  - A különböző lokális számítógépeket a portokban kódoljuk
  - Ezeket a WAN-hoz csatlakozó router megfelelően átszámítja
  - Kimenő csomagoknál a LAN-IP-cím és egy kódolt Port kerül megadásra mint forrás
  - Érkező csomagoknál (melyeknek a célja a LAN-IP-cím), a kódolt Port alapján a lokális számítógép és a hozzátartozó Port egy táblázat segítségével számítható vissza

## NAT és PAT -- előnyök

- Előnyök
  - A lokális hálózat számítógépei direkt nem elérhetők
  - Megoldja/enyhíti az IPv4 címek szűkösségének a problémáját
  - Lokális számítógépek nem szolgálhatnak szerverként
- DHCP (Dynamic Host Configuration Protocol)
  - Hasonló előnyöket biztosít

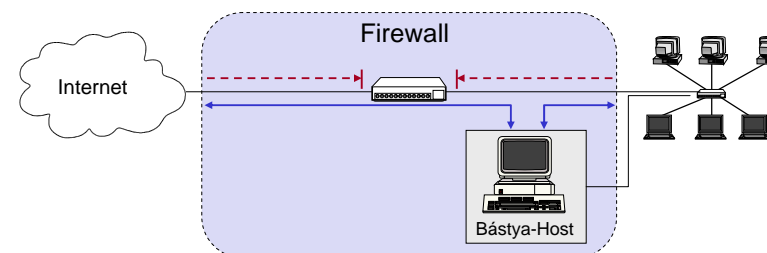
## Tűzfal architektúra – egyszerű csomagszűrő

- Realizálható
  - egy standard workstation (pl. Linux PC) által, amely két hálózati interfésszel és szűrő szoftverrel rendelkezik vagy
  - speciális, szűrésre képes router által



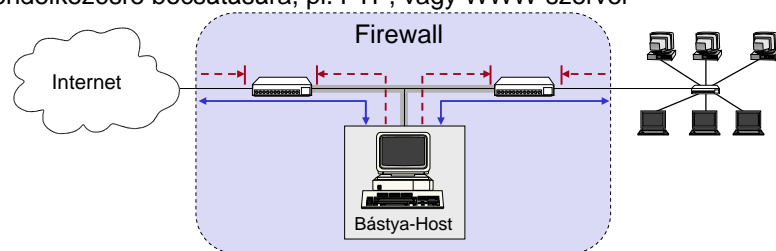
## Tűzfal architektúra -- Screened Host

- A csomagszűrő
  - csak az Internet és a screened host között és a
  - screened host és a védett hálózat között enged meg forgalmat
- A screened host proxy szolgáltatást kínál fel
  - A screened host bástya-hostként működik, képes önmaga támadást elhárítani



## Tűzfal architektúra -- Screened Subnet

- Perimeter hálózat két csomagszűrő között
- A belső csomagszűrő védi a belső hálózatot, ha a bástya-hostnak nehézségei támadnak
  - Egy hackelt bástya-host így nem tudja a belső hálózati forgalmat kikémlelni
- Perimeter hálózatok különösen alkalmasak nyilvános szolgáltatások rendelkezésre bocsátására, pl. FTP, vagy WWW-szerver



## Tűzfal -- csomagszűrő

- Mit lehet elérni csomagszűrőkkel
  - Elvileg majdnem mindent, mert a teljes kommunikáció csomagokkal történik...
  - Gyakorlatban hatékonysági kérdéseket kell mérlegelni egy proxy-megoldással szemben
- Alap csomagszűrés lehetővé teszi az adatátvitel ellenőrzését a következők alapján
  - Source IP Address
  - Destination IP Address
  - Transport protocol
  - Source/destination application port
- Csomagszűrés (és tűzfalak) korlátai
  - Tunnel algoritmusok nem ismerhetők fel
  - Lehetséges betörni más kapcsolatok által is
    - pl. Laptop, UMTS, GSM, Memory Stick

## Felhasználói réteg

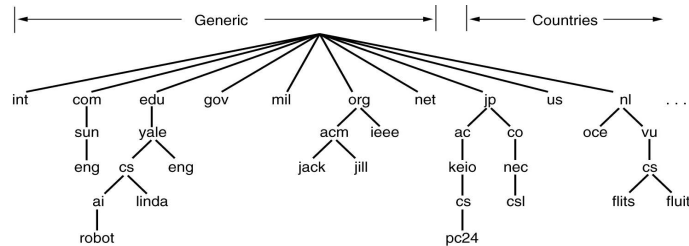
- Domain Name System
- Példák a felhasználói rétegre:
  - E-Mail
  - WWW
  - Content Delivery Networks
  - Peer-to-Peer-Networks
- A forgalom az Interneten

## Domain Name System (DNS)

- Az emberek számára 4 byte IPv4 cím nehezen kezelhető:
  - 209.85.135.99 google.com-hoz
  - 157.181.151.154 az ELTE-hez
  - Mit jelent?
    - 207.46.19.30
    - 157.181.35.45
- Jobb: Természetes szavak az IP-címekhez
  - Pl. www.google.com
  - vagy www.elte.hu
- A Domain Name System (DNS)
  - lefordítja ezeket a címeket IP-címekre (és fordítva)
  - elosztott adatbázis

## DNS – Felépítés

- DNS neveket képez le IP-címekre
  - Pontosabban: neveket erőforrás-bejegyzésekre
- A nevek hierarchikusan struktúráltak egy névtérben
  - Max. 63 jel komponensenként, összesen max. 255 jel
  - Minden domain-en belül, a domain tulajdonosa ügyeli fel a névteret a domain alatt



## DNS Resource Record

- **Erőforrás bejegyzés** (resource record RR): a domain-ekről, egyes host-okról, stb... adnak információt

- RR formátum: (name, ttl, class, type, value)

- name: pl. domain név vagy host név
- ttl (time to live): érvényesség (másodpercben)
- class: Internet esetén mindig "IN"
- type: lásd a táblázatot
- value: pl. IP-cím

Type	Meaning	Value
SOA	Start of Authority	Parameters for this zone
A	IP address of a host	32-Bit integer
MX	Mail exchange	Priority, domain willing to accept e-mail
NS	Name Server	Name of a server for this domain
CNAME	Canonical name	Domain name
PTR	Pointer	Alias for an IP address
HINFO	Host description	CPU and OS in ASCII
TXT	Text	Uninterpreted ASCII text

- RR Példa:  
pandora.inf.elte.hu. 43200 IN A 157.181.161.52

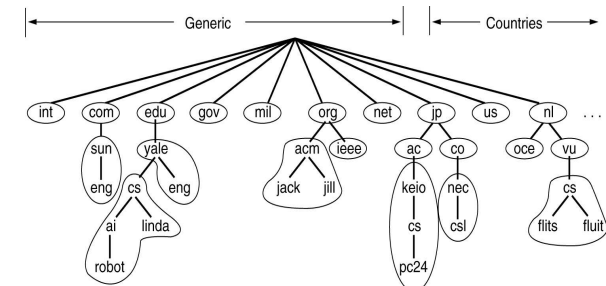
## DNS Resource Records -- Példák

Példák RR típusokra

- Type=A
  - name: egy végrendszer (host) neve
  - value: egy IP-cím
- Type=NS
  - name: egy domain (pl elte.hu)
  - value: a domain authoritative name server-jének az IP-címe
- Type=MX
  - value: a name-hez tartozó mail server neve
- Type=CNAME
  - name: egy alias név egy kanonikus névhez
  - value: a kanonikus név
- Type = SOA (start of authority)
  - name: a domain neve
  - value: szerverek neve, melyek a zónához tartozó mérvadó információkat rendelkezésre bocsátják, paraméterek a zónához
    - a zóna sorszám, a zóna sorszáma,
    - frissítési intervallum a másodlagos szervernek,...

## DNS Name Server

- A névtér **zónákra** van osztva
- Minden zónához tartozik egy **Authoritativ Server** a mérvadó információval
  - Egy **Primary Name Server**
  - Továbbá egy vagy több **Secondary Name Server** a megbízhatóság miatt
- Minden Name Server ismeri
  - a saját zónáját
  - a gyermek-zónák Name-Server-jeit

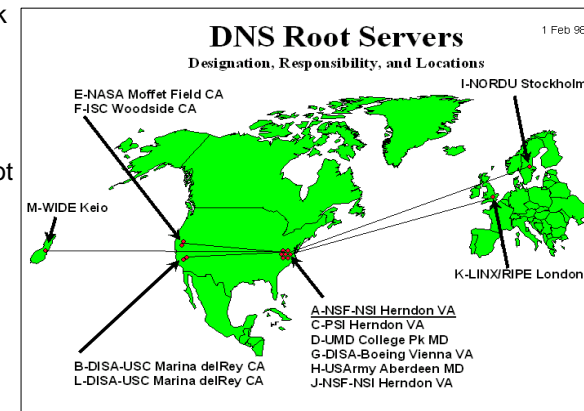


## Servers/Resolvers

- Minden végrendszernek van egy „feloldója” (resolver)
  - Tipikusan egy könyvtár, amit felhasználásokhoz kapcsolhatunk
  - Lokális name-server-ek kézzel konfigurálva (pl. /etc/resolv.conf)
- Name servers
  - Tipikusan egy zónáért felelősek
  - Lokális szerverek
    - A lokális végrendszereknek végznek lekérdezéseket távoli végrendszer nevekről
    - Megválaszolják a lekérdezéseket a lokális zónáról

## DNS: Root Name Servers

- A “root” zónáért felelősek
- Jelenleg 13 root name server világszerte
  - A-M „számozva”
- Lokális szerverek kapcsolatba lépnek a root szerverrel, ha ők nem tudják megválaszolni a lekérdezést
  - Jól ismert root szerverekkel konfiguráltak



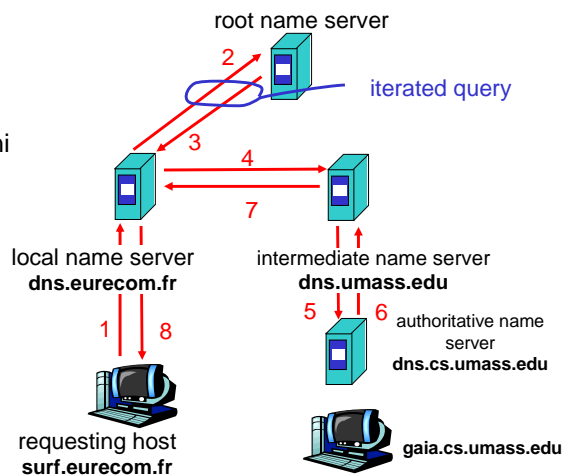
## DNS lekérdezések

### Iteratív lekérdezés:

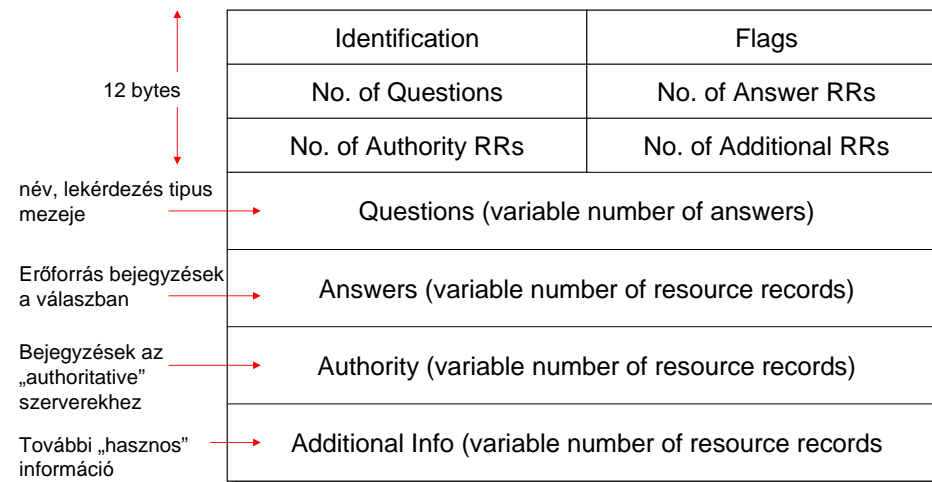
- A megkérdezett szerver annyi információt ad a válaszban, amit ő maga tud
- Pl. annak a szervernek a nevét, akit meg kell kérdezni

### Rekurzív lekérdezés:

- A megkérdezett szerver rekurzív „kideríti” a hiányzó információt
- A lokális szerverek tipikusan rekurzív lekérdezési módban dolgoznak
- Root vagy távoli szerverek iteratívban



## DNS üzenet formátum



## Tipikus feloldási folyamat

- A `www.inf.elte.hu` név feloldásának lépései
  - A felhasználás hívja a `gethostbyname()` függvényt
  - A végrendszer lekérdezi a lokális name server-t ( $S_1$ )
  - $S_1$  lekérdezi a root server-t ( $S_2$ ) a `www.inf.elte.hu` névvel
  - $S_2$  válaszol a `elte.hu`-hoz ( $S_3$ ) tartozó NS bejegyzéssel
  - Honnan tudjuk meg az A bejegyzést  $S_3$ -hoz
    - Erre való az „additional information section”
  - $S_1$  lekérdezi  $S_3$ -t a `www.inf.elte.hu` névvel
  - $S_3$  válaszol a `www.inf.elte.hu`-hoz tartozó A bejegyzéssel
- Több A bejegyzés is érkezhethet a válaszban → mit jelent ez?

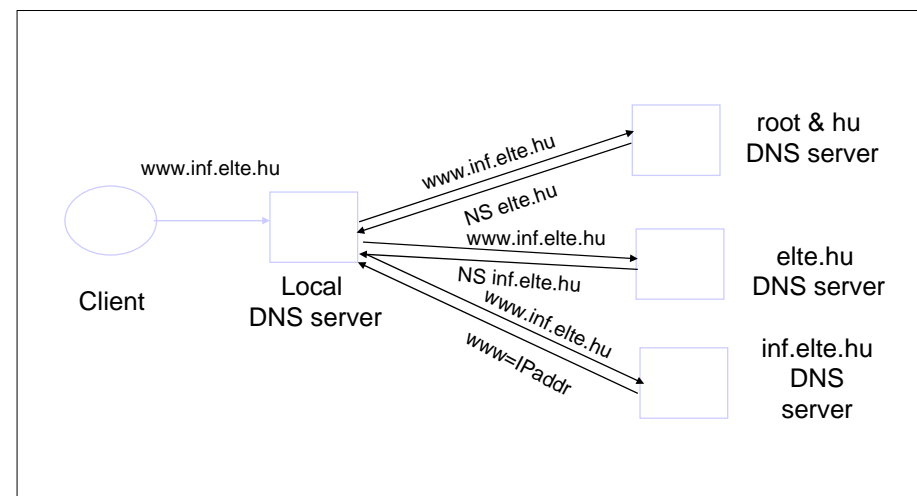
## Caching

- DNS válaszok tárolódnak az érintett szervereken (caching)
  - Gyors válasz ismételt lekérdezés esetén
  - Más lekérdezések bizonyos részeket újra felhasználhatnak a válaszból
    - PI. NS bejegyzéseket a domain-ekhez
- DNS negatív lekérdezések tárolódnak a cache-ben
  - Ne kelljen megismételni a kudarcot
  - PI. elgépelés
- A cache-ben tárolt adatok érvényessége egy idő után lejár
  - Az érvényesség idejét (TTL) az adat tulajdonosa határozza meg
  - Minden bejegyzés tartalmaz TTL-t

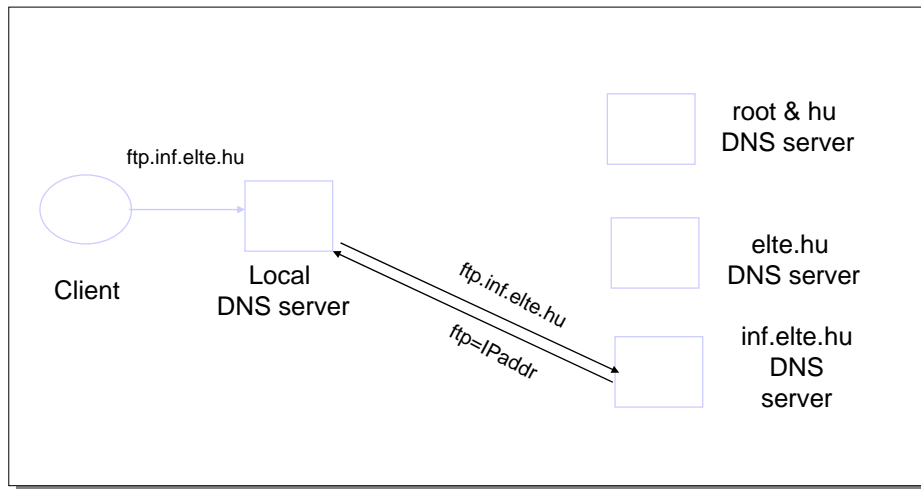
## Prefetching

- Name server minden válaszhoz adhat további adatokat
- Tipikusan prefetching-hez használják
  - CNAME/MX/NS tipikusan más végrendszer nevére mutat
  - Válaszok tartalmazzák a végrendszerek címét, amelyekre mutatnak az “additional section” részben

## DNS lekérdezés példa



## Példa egy későbbi lekérdezésre

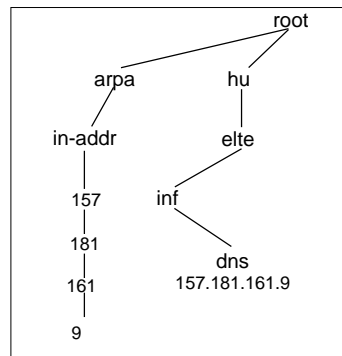


## Megbízhatóság, rendelkezésre állás

- DNS szerverek replikáltak
  - A name service működik, ha egy replika működik
  - A lekérdezések kiegyensúlyozhatók a replikák között (load balancing)
- UDP-t használ a lekérdezéshez
  - Megbízhatónak kell lenni → Miért nem TCP?
  - Timeout esetén alternatív szervert próbál
  - „Exponential backoff”, ha visszatér ugyanahhoz a szerverhez
  - Ugyanaz az azonosító minden lekérdezéshez
    - Mindegy melyik szerver válaszol

## Reverse Name Lookup

- Melyik számítógéphez tartozik az 157.181.161.9 IP-cím?
  - Lekérdezés: 9.161.181.157.in-addr.arpa
  - Miért van megfordítva a cím?
  - dns.inf.elte.hu



## Dinamikus DNS

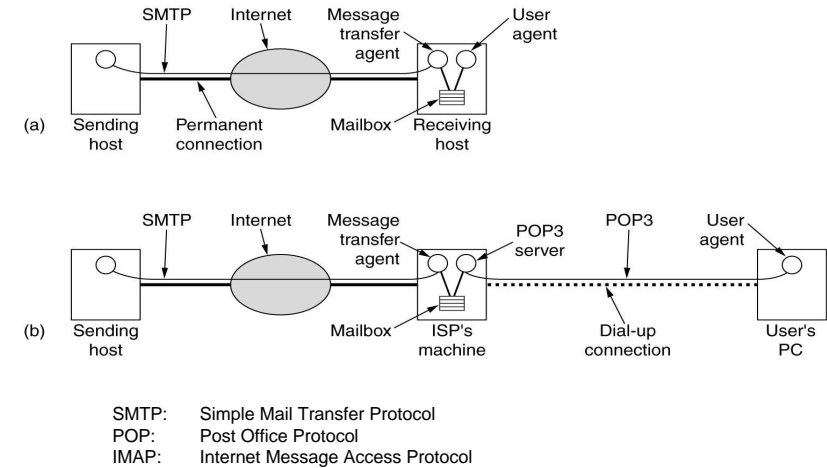
- Probléma
  - Időlegesen hozzárendelt IP-címek
  - Pl. DHCP által
- Dinamikus DNS
  - Amint egy csomópont egy új IP-címet kap, regisztrálja azt azon a DNS-szerveren, amely őérte felelős
  - Rövid TTL bejegyzések biztosítják azt, hogy a bejegyzések gyorsan aktualizálódjanak
    - egyébként a lekérdezések rossz számítógépre irányítódnának
- Felhasználás
  - Egy privát domain regisztrálása
  - lásd [www.dyndns.com](http://www.dyndns.com)



## Email (RFC 821/822)

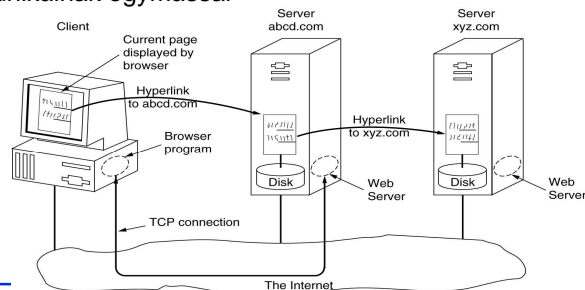
- **Komponensei:**
  - user agents (UA)
  - message transfer agents (MTA)
- **Szolgáltatások**
  - kompozíció, küldés, értesítés, megjelenítés, rendelkezés (disposition)
- **További szolgáltatások**
  - továbbküldés, auto-válasz, szabadság-funkciók, levelező listák, ...
- **Struktúra:**
  - Boríték – a szállításhoz szükséges információ, a MTA használja
  - Tartalom
    - Fejléc – kontroll információ a UA-nek
    - Törzs – a valódi tartalom

## E-Mail: SMTP és POP



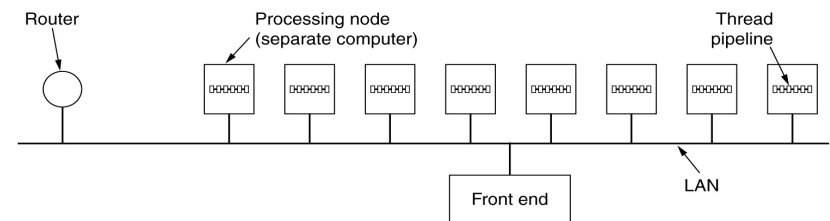
## World Wide Web

- **Client-Server-Architektúra**
  - **Web-Server** web-oldalakat bocsát rendelkezésre
  - Formátum: **Hypertext Markup Language (HTML)**
  - **Web-Browser** oldalakat kérdez le a web-server-től
  - Server és browser **Hypertext Transfer Protocol (HTTP)** által kommunikálnak egymással



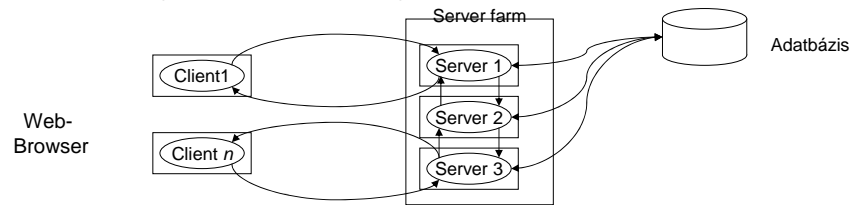
## Szerver-Farm

- **A szerver oldal teljesítményének növeléséhez**
  - több web-server dolgozik
- **Front end**
  - Fogadja a lekérdezéseket
  - Továbbítja a lekérdezéseket egy különálló csomóponthoz további feldolgozásra



## Web-Server-ek és adatbázisok

- Web-Server-ek nem csak statikus web-oldalakat bocsátanak rendelkezésre
  - Web-oldalakat automatikusan is létre lehet hozni
  - Ehhez egy adatbázisból kérdeznék le adatokat
  - Ez az adatbázis nem szükségszerűen statikus, interakció által megváltoztatható lehet
- Probléma:
  - **Konzisztencia**
- Megoldás
  - Web-szolgáltatás és adatbázis egy 3-fokú architektúrája



## Web-Cache

- Server-Farm ellenére a várakozási idő gyakran kritikus
- Megoldás:
  - Cache (Proxy)
- Helye
  - A kliens oldalon
  - A lokális hálózatban (egy Proxy-n)
  - Az Internet-Service-Provider-nél
- Kérdések
  - Adatok elhelyezése, nagysága, aktualitása
  - Érvénytelenítés Time-Out által

## Content Distribution Networks (CDN)

- Cache-ek koordinált halmaza
  - Nagy web-helyek terhelését elosztja globálisan elosztott szerver-farmon
  - Lehetőleg különböző szervezetek web-oldalainak kezelése
    - pl. hírek, szoftver-gyártók, kormányok
  - Példák: Akamai, Digital Island
  - A Cache-lekérdezések regionálisan és terhelést tekintve a leginkább megfelelő helyre kerülnek átirányításra
- Példa Akamai:
  - Elosztott hash-tábla által lehetséges az oldalak/adatok elosztása hatékonyan és lokálisan