

# Számítógépes Hálózatok 2011

## 12. Szállítói réteg – TCP, hatékonyság, fairness; Biztonság

### Torlódás elkerülési elv: AIMD

- A TCP a „fast recovery” mechanizmussal lényegében a következőképp viselkedik:

x: csomagok száma per RTT

- Kapcsolatfelépítés:

$$x \leftarrow 1$$

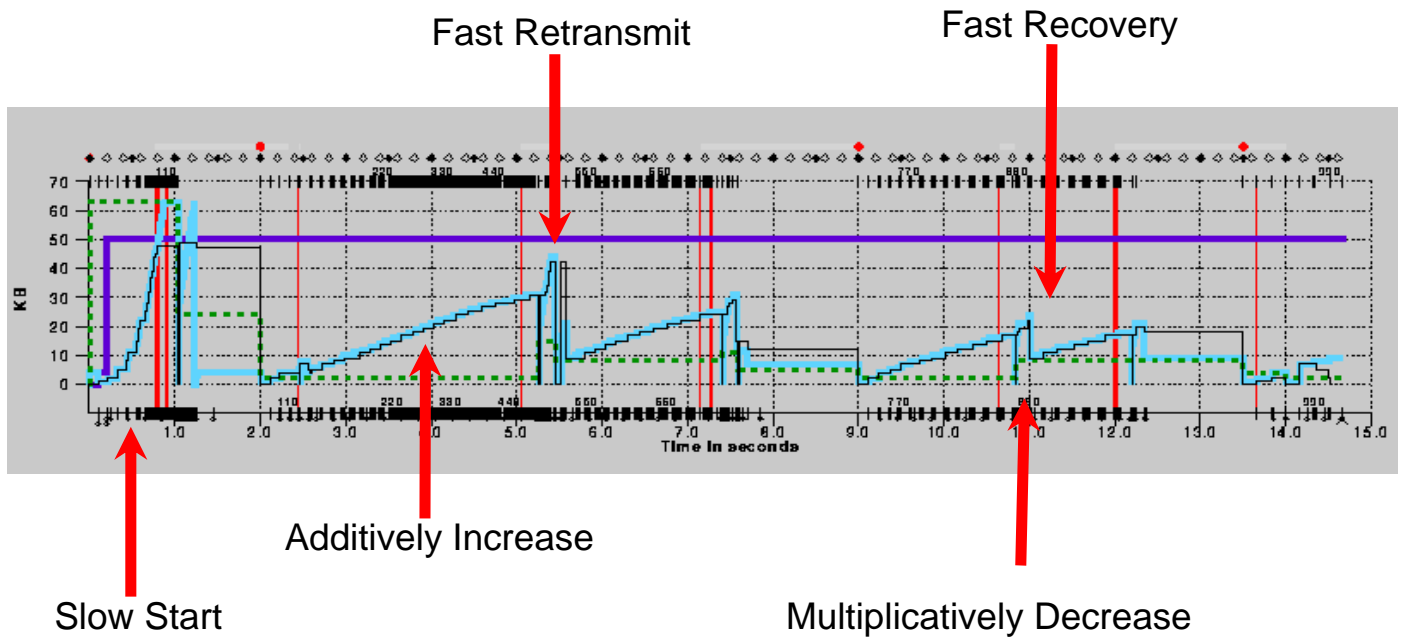
- Csomagvesztésnél, MD: multiplicative decreasing

$$x \leftarrow x/2$$

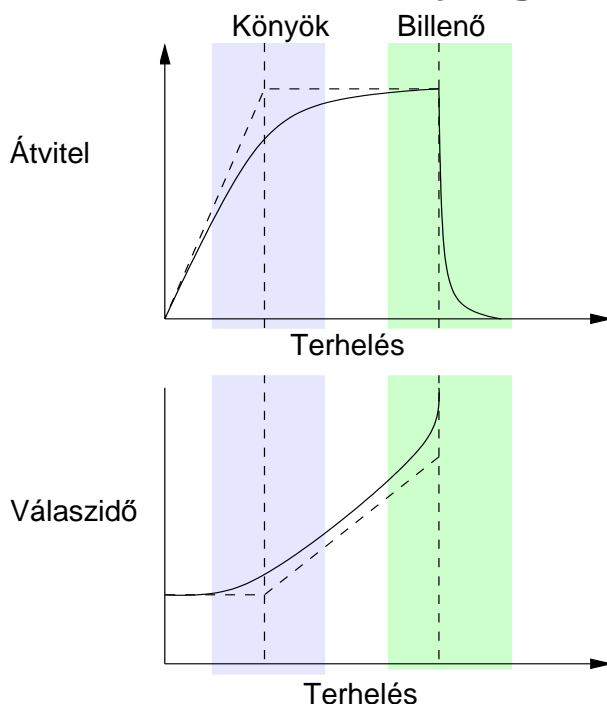
- Nyugtázott szegmenseknél, AD: additive increasing

$$x \leftarrow x + 1$$

## Példa: TCP Reno „in akción”



## Additive Increase Multiplicative Decrease (AIMD): Fairness és Hatékonyság



A hálózati terhelés az átvitelrel és a válaszával kölcsönösen hat egymásra.

- Az átvitel maximális, ha a terhelés a hálózat kapacitását majdnem eléri.
- Ha a terhelés tovább nő, túlszordulnak a pufferek, csomagok vesznek el, újra kell küldeni, drasztikusan nő a válaszá. Ezt a toródást **congestion**-nak nevezzük.
- Ezért a maximális terhelés helyett, ajánlatos a hálózat terhelését a könyök közelében beállítani. Itt a válaszá csak lassan emelkedik, míg az adatátvitel már a maximum közelében van.
- Egy jó torlódáselkerülési (*congestion avoidance*) stratégia a hálózat terhelését a könyök közelében tartja: **hatékonyság**. Emellett fontos, hogy minden résztvevőt egyforma rátával szolgáljunk ki: **fairness**.

## AIMD Fairness és Hatékonyság – Egy egyszerű modell

- n résztvevő, forduló-modell
- résztvevő  $i$  adatrátája a  $t$ -edik fordulóban  $x_i(t)$
- Kezdeti adatráták:  $x_1(0), \dots, x_n(0)$
- A visszacsatolás (feedback) forduló  $t$  után:  $y(t) = 0$ , ha  $\sum_{i=1}^n x_i(t) \leq K$

$$y(t) = 1, \text{ ha } \sum_{i=1}^n x_i(t) > K$$

- Minden résztvevő aktualizálja az adatrátáját a  $t+1$ -edik fordulóban:

$$x_i(t+1) = f(x_i(t), y(t))$$

- Increase-stratégia  $f_0(x) = f(x, 0)$
  - Decrease-stratégia  $f_1(x) = f(x, 1)$
- Tekintsük a következő lineáris függvényeket :

$$f_0(x) = a_I + b_I x, \quad f_1(x) = a_D + b_D x$$

## AIMD Fairness és Hatékonyság– A Modell

- A következő lineáris függvényeket vizsgáljuk:

$$f_0(x) = a_I + b_I x, \quad f_1(x) = a_D + b_D x$$

- Érdekes speciális esetek:

- MIMD: Multiplicative Increase/Multiplicative Decrease

$$f_0(x) = b_I x, \quad f_1(x) = b_D x, \quad \text{ahol } b_I > 1, b_D < 1.$$

- AIAD: Additive Increase/Additive Decrease

$$f_0(x) = a_I + x, \quad f_1(x) = a_D + x, \quad \text{ahol } a_I > 0, a_D < 0.$$

- AIMD: Additive Increase/Multiplicative Decrease

$$f_0(x) = a_I + x, \quad f_1(x) = b_D x, \quad \text{ahol } a_I > 0, b_D < 1.$$

## AIMD Fairness és Hatékonyság

- Hatékonyság

- Terhelés:  $X(t) := \sum_{i=1}^n x_i(t)$

- Mérték:  $|X(t) - K|$

- Fairness:  $x=(x_1, \dots, x_n)$  esetén:

$$F(x) = \frac{(\sum_{i=1}^n x_i)^2}{n \sum_{i=1}^n (x_i)^2} .$$

- $1/n \leq F(x) \leq 1$
  - $F(x) = 1 \leftrightarrow$  absolut Fairness
  - skálázástól független
  - Folytonos, differenciálható
  - Ha  $n$  közül  $k$  fair, a többi 0, akkor  $F(x) = k/n$

## Konvergencia

- Konvergencia nem lehetséges

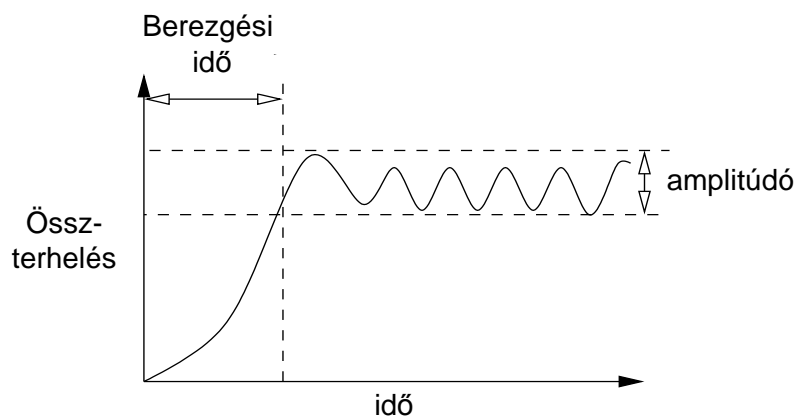
- Legjobb esetben oszcilláció az optimális érték körül

- Az oszcilláció amplitúdója  $A$

$$A = \inf_{t_0 \geq 0} \sup_{t \geq t_0} |X(t) - K| .$$

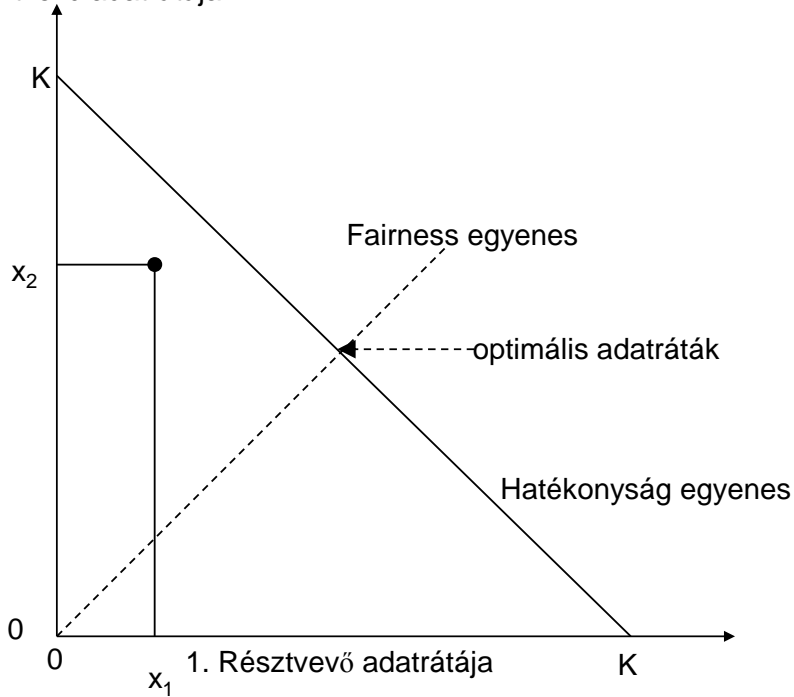
- Berezgési idő  $T$

$$T = \min\{t_0 \mid \forall t \geq t_0 : |X(t) - K| \leq A\} .$$



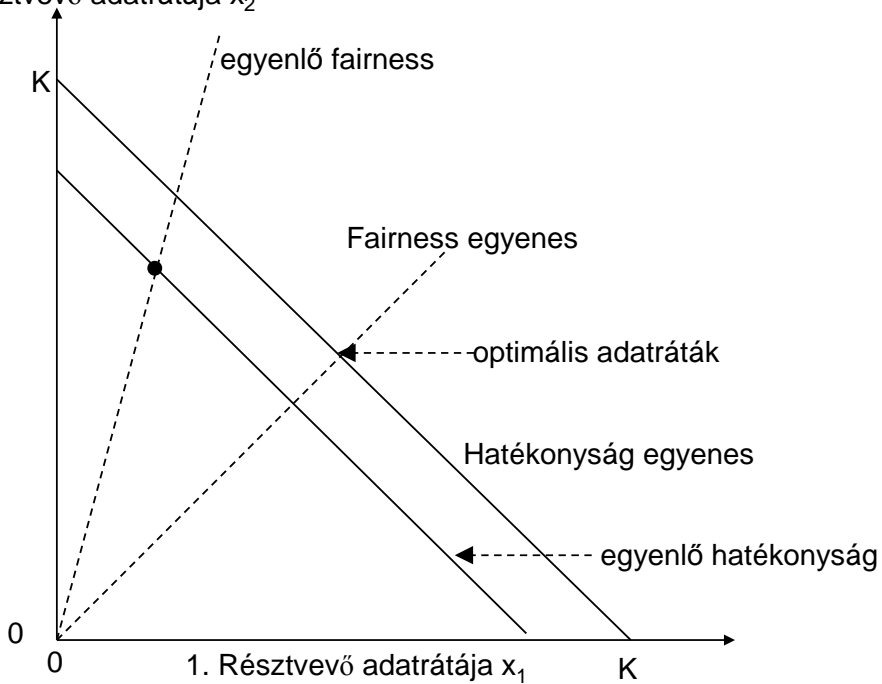
# Vektor Ábrázolás (I)

2. Résztevő adatrátája

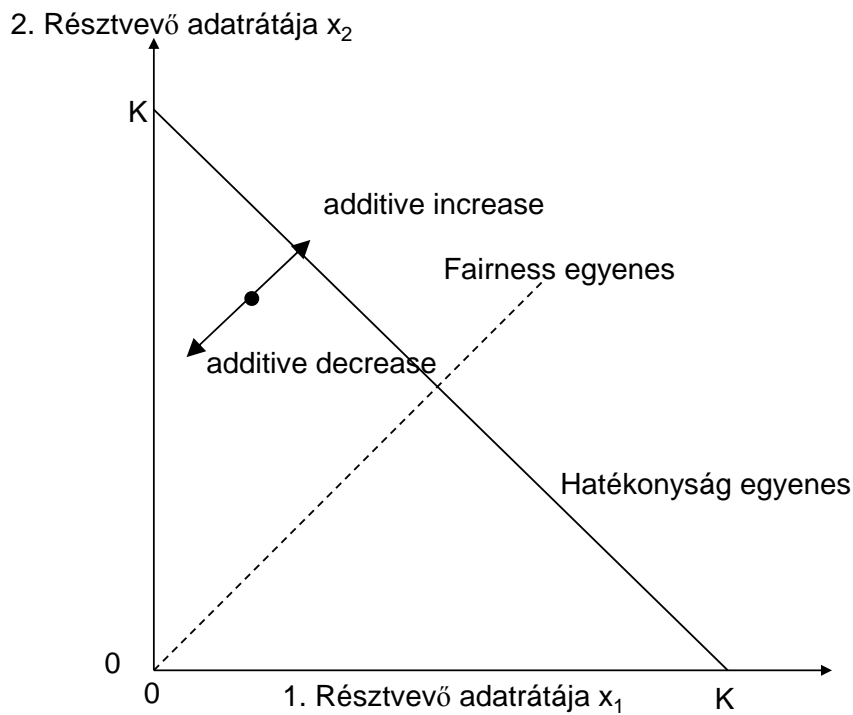


# Vektor Ábrázolás (I)

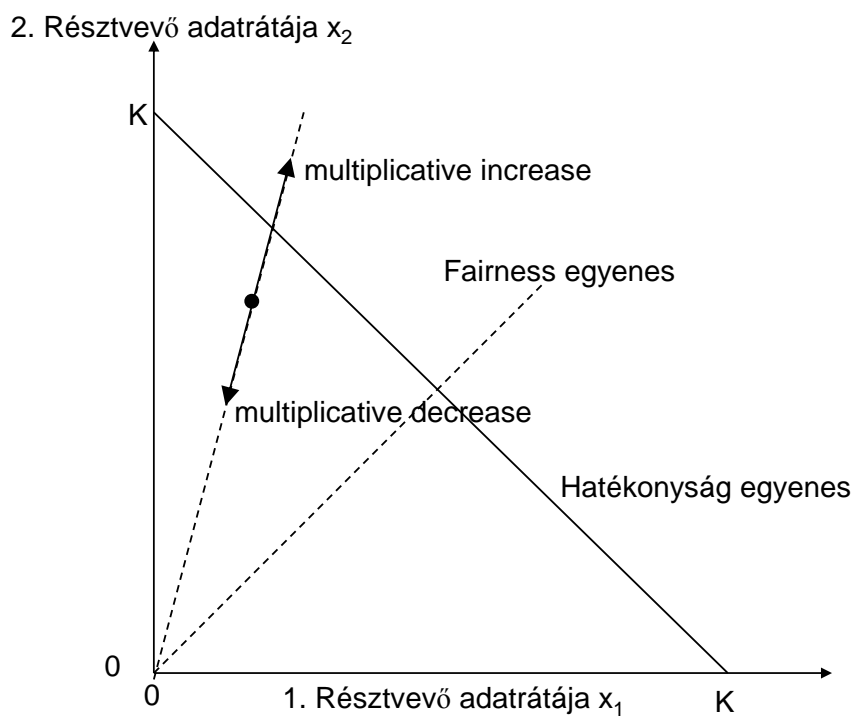
2. Résztevő adatrátája  $x_2$



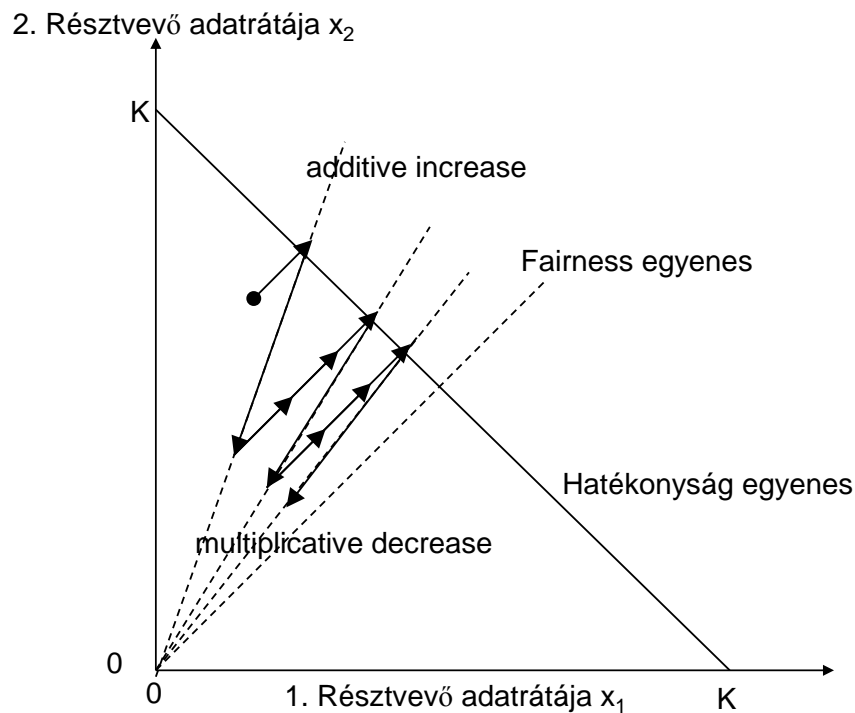
## Vektor Ábrázolás (I)



## Vektor Ábrázolás (I)



## Vektor Ábrázolás (I)



## TCP összefoglalás

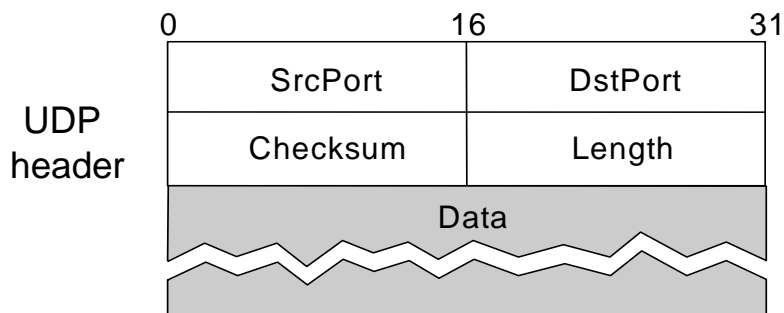
- TCP egy megbízható byte-folyamot hoz létre
  - Hibafelügyelet "Go-Back-N" által
- Congestion control
  - Ablak alapú
  - AIMD, Slow start, *Congestion Threshold*
  - Folyamfelügyelet *Window* által
  - Kapcsolatfelépítés
  - Nagle algoritmus

## TCP fairness & TCP friendliness

- TCP
  - Dinamikusan reagál a rendelkezésre álló sávszélességre
  - A sávszélesség fair felosztása
    - Ideális esetben:  $n$  TCP-kapcsolat mindegyike  $1/n$  részt kap
- TCP más protokollokkal
  - Reakció más szállítói protokollok terhelésétől függ
    - pl. UDP-ben nincs congestion control
  - Más protokollok mindenkor felhasználhatók
  - UDP és más protokoll el tudja nyomni a TCP kapcsolatokat
- Véggöveztetés
  - A szállítói protokolloknak TCP-kompatibilisnek kell lenni (TCP friendly)

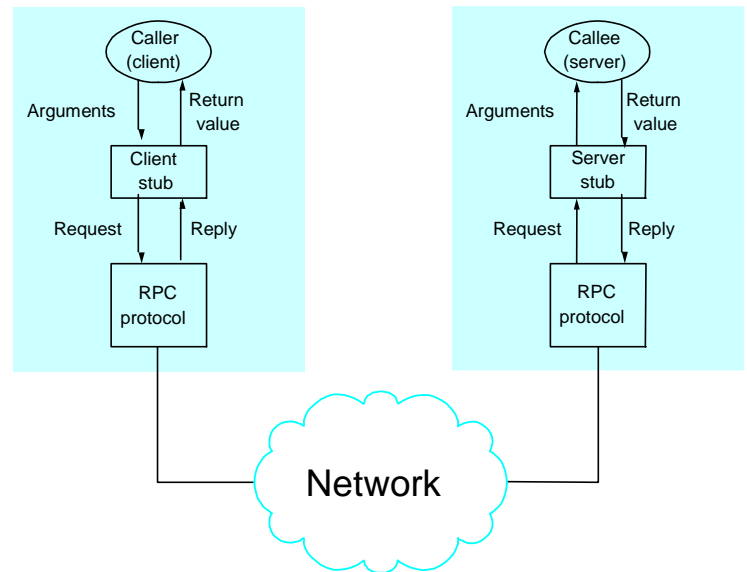
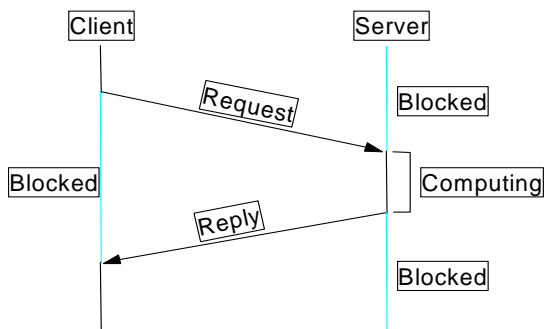
## UDP

- User Datagram Protocol (UDP)
  - Egy nem megbízható kapcsolat nélküli szállítói réteg protokoll csomagoknak
- Fő funkció:
  - A hálózati réteg csomagjainak demultiplexálása
- Egyéb funkció (opcionális):
  - Checksum: UDP header + data





## Remote procedure call – Struktúra



- Komplexebb interakcióhoz:
  - Egy függvény hívása egy másik számítógépen
- Cél: Transzparens protokoll a hívónak/hívottnak

## Számítógéphálózatok biztonsága

- Szerepet játszik a következő rétegekben
  - Fizikai réteg
  - Adatkapcsolati réteg
  - Hálózati réteg
  - Szállítói réteg
  - Alkalmazási réteg
- Mi jelent fenyegetést (vagy támadást)
- Milyen módszerek vannak
  - Kriptográfia
- Hogyan védekezhetünk támadások ellen?
  - Példa: Firewalls

## Számítógéphálózatok biztonsága

## Fenyegetés, támadás

- Definíció:
  - Egy számítógéphálózat fenyegetése minden olyan lehetséges esemény vagy akciók sorozata, amely biztonsági célok megsértéséhez vezethet
  - A támadás a fenyegetés realizálása
- Példa:
  - Egy hacker betör egy zárt hálózatba
  - Az átfutó email-ek nyilvánosságra hozása
  - Idegen hozzáférés egy online bankszámlához
  - Egy hacker egy rendszer összeomlását okozza
  - Valaki autorizálatlanul tevékenykedik valaki más nevében (Identity Theft)

## Biztonsági célok

- Bizalmaság (confidentiality):
  - Csak egy előre meghatározott publikum tudja írni vagy olvasni az átvitt vagy tárolt adatokat
  - A résztvevők azonosságának a bizalmassága: Anonimitás
- Adatintegritás (data integrity)
  - Adatok megváltoztatása kideríthető legyen
  - Az adatok szerzője felismerhető legyen
- Felelős hozzárendelhetősége (accountability)
  - Minden kommunikációs eseményhez hozzárendelhető legyen annak okozója
- Rendelkezésre állás (availability)
  - A szolgáltatások elérhetőek legyenek és helyesen működjenek
- Kontrollált hozzáférés (controlled access)
  - A szolgáltatásokat és az információkat csak autorizált felhasználók érik el

## Támadások technikai definíciója

- Álarc (masquerade)
  - Valaki másnak adja ki magát
- Lehallgatás (eavesdropping)
  - Valaki olyan információt olvas, amit nem neki szántak
- Hozzáférési jog megsértése (Authorization Violation)
  - Valaki olyan szolgáltatást vagy erőforrást használ, ami nem neki szánt
- (Átvitt) Információ elvesztése vagy megváltoztatása
  - Az adatokat megváltoztatják vagy megsemmisítik
- Kommunikáció letagadása (denial of communication acts, repudiation)
  - Valaki (hamisan) letagadja a részvételét a kommunikációban
- Információ hamisítás (forgery of information)
  - Valaki más nevében állít elő (változtat) üzeneteket
- Sabotage
  - Minden olyan akció, amely a szolgáltatások vagy a rendszer helyes működését vagy rendelkezésre állását csökkenti

## Fenyegetések és biztonsági célok

biztonsági cél	fenyegetés						
	álarc	lehallgatás	hozzáférési jog megsértése	információ elvesztése vagy megváltoztatása	kommunikáció letagadása	információ hamisítás	sabotage (pl. túlterhelés)
bizalmasság	x	x	x				
adatintegritás	x		x	x		x	
felelős hozzáférhetőség	x		x		x	x	
rendelkezésre állás	x		x	x			x
kontrollált hozzáférés	x		x			x	

## A kommunikációs biztonság terminológiája

- Biztonsági szolgáltatás
  - Egy absztrakt szolgáltatás, amely egy biztonsági tulajdonságot kíván biztosítani
  - Lehet kriptografikus protokollal vagy anélkül realizálni, pl.
    - Adatok titkosítása egy merev lemezen
    - CD a páncélszekrényben
- Kriptografikus algoritmus
  - matematikai transzformáció
  - kriptografikus (titkosító) protollokban használt
- Kriptografikus protokoll
  - lépések és kicsírelendő üzenetek sora egy biztonsági cél eléréséhez

## Biztonsági szolgáltatás

- Authentifikáció
  - Digitális aláírás: az adat bizonyíthatóan a létrehozótól származik
- Integritás
  - Biztosítja, hogy az adat ne legyen észrevétel nélkül megváltoztatható
- Bizalmasság
  - Az adat csak a fogadó által érthető
- Kontrollált hozzáférés
  - Biztosítja, hogy csak az arra jogosultak férjenek hozzá a szolgáltatásokhoz és információkhoz
- Letagadhatatlanság
  - Bizonyítja, hogy az üzenet letagadhatatlanul az előállítójától származik

## Kriptológia

- Kriptológia
  - A titkos kommunikáció tudománya
  - A görög kryptós (rejtett) és lógos (szó) szavakból
  - Kriptológia részei:
    - Kriptográfia (gráphein = írás): titkos kommunikáció létrehozásának a tudománya
    - Kripto-analízis (analýein = megoldani, kibogozni): titkosított információ kibogozásának a tudománya

## Titkosítási módszerek

- Szimmetrikus titkosítási módszerek
  - pl. Caesar kód
  - Enigma
  - DES (Digital Encryption Standard)
  - AES (Advanced Encryption Standard)
- Kriptografikus Hash-függvények
  - SHA-1, SHA-2, MD5
- Aszimmetrikus titkosítási módszerek
  - RSA (Rivest, Shamir, Adleman)
  - Diffie-Helman
- Digitális aláírás
  - PGP (Phil Zimmermann), RSA

## Szimmetrikus titkosítási módszerek

- pl. Caesar kód, DES, AES
- Léteznek  $f, g$  függvények, úgy hogy
  - titkosítás:
    - $f(\text{kulcs}, \text{szöveg}) = \text{kód}$
  - visszakódolás:
    - $g(\text{kulcs}, \text{kód}) = \text{szöveg}$
- A kulcsnak
  - titokban kell maradni
  - a küldő és a fogadó számára ismertnek kell lenni

## Kriptografikus hash-függvények

- pl. SHA-1, SHA-2, MD5
- Egy kriptografikus hash-függvény  $h$  egy szöveget képez le egy fix hosszúságú kódra, úgy hogy
  - $h(\text{szöveg}) = \text{kód}$
  - és nincs olyan másik szöveg  $\text{szöveg}'$ , melyre:
    - $h(\text{szöveg}') = h(\text{szöveg})$  és  $\text{szöveg} \neq \text{szöveg}'$
- Lehetséges megoldás:
  - Szimmetrikus kriptografikus módszerek felhasználása

## Aszimmetrikus titkosítási módszerek

- pl. RSA, Ronald Rivest, Adi Shamir, Lenard Adleman, 1977
  - Diffie-Hellman, PGP
- Privát kulcs `privat`
  - titkos, csak az üzenet fogadója ismeri
- Nyilvános kulcs `public`
  - minden résztvevő ismeri
  - egy függvény állítja elő
    - $\text{keygen}(\text{privat}) = \text{public}$
- Titkosító függvény  $f$  és visszakódoló függvény  $g$ 
  - mindenki számára ismert
- Titkosítás
  - $f(\text{public}, \text{text}) = \text{code}$
  - minden résztvevő ki tudja számítani
- Visszakódolás
  - $g(\text{privat}, \text{code}) = \text{text}$
  - csak a fogadó tudja kiszámítani

## Példa: RSA

- Az eljárás a prím-faktor felbontás nehézségére alapul

- 1. példa:  $15 = ? * ?$

- $15 = 3 * 5$

- 2. példa:

3865818645841127319129567277348359557444790410289933586483552047443

=

$$1234567890123456789012345678900209 * \\ 313131313131313131313131313131300227$$

- Máiig nem ismert hatékony eljárás a prím-faktor felbontásra
  - De két prím szorzata hatékonyan kiszámítható
  - Prím számok hatékonyan meghatározhatók
  - Prím számok gyakoriak

## Az RSA-séma

1. Legyen  $p, q$  két nagy prím szám (1024-2048 bit)
2. Számítsuk ki  $n = p q$
3. Számítsuk ki  $\phi(n) = (p-1)(q-1)$  (Euler  $\phi$  függvény)
4. Legyen  $e$  egy szám,  $1 < e < \phi(n)$ , úgy hogy  $e$  és  $\phi(n)$  relatív prím
5. Legyen  $d$  egy szám, melyre  $e d = 1 \pmod{\phi(n)}$

- Privát kulcs  $(n, d)$

- Nyilvános kulcs:  $(n, e)$

- Visszakódolás:

- $message = code^d \pmod n$

- Titkosítás:

- $code = message^e \pmod n$

- Euler tétele:

- $\forall m$  egészre,  $m$  és  $n$  relatív prím:  $m^{\phi(n)} = 1 \pmod n$

- Helyesség (ha  $message$  és  $n$  relatív prím):

$$(message^e \pmod n)^d \pmod n \\ = message^{e d} \pmod n = message^{e d \pmod{\phi(n)}} \pmod n \\ = message \pmod n$$



## Az RSA-séma

- RSA-séma helyessége:

$$\begin{aligned} & \text{code}^d \bmod n \\ &= (\text{message}^e \bmod n)^d \bmod n \\ &= \text{message}^{e \cdot d} \bmod n \end{aligned}$$

Mivel  $ed = 1 \bmod (p-1)(q-1)$  és így  
 $ed = 1 \bmod (p-1)$  és  
 $ed = 1 \bmod (q-1)$ ,

a kis Fermat tétel (változat) alapján  
 $\text{message}^{ed} = \text{message} \bmod p$

és

$$\text{message}^{ed} = \text{message} \bmod q$$

Ekkor a Kínai maradék tétel miatt

$$\text{message}^{ed} = \text{message} \bmod pq$$

- Kis Fermat tétel:

- $\forall p$  prímszámra és  $m$  egészre:  $m^p = m \bmod p$

- Változat:

- $\forall p$  prímszámra,  $m$  egészre, ha  $i, j$  pozitív egészek,  $i \equiv j \bmod p-1$ , akkor  
 $m^i = m^j \bmod p$

- Kínai maradék tétel:

- $\forall n_1, n_2, \dots, n_k$  egészre, melyek páronként relatív prímszámok, és  $\forall a_1, a_2, \dots, a_k$  egészre  
 $\exists x$  egész, melyre  $x = a_i \bmod n_i, i=1, \dots, k$ .  
Továbbá minden ilyen  $x$  kongruens  
moduló  $n=n_1 n_2 \dots n_k$ .

## RSA példa

- Két nagy prímszám 7, 11

- $n=77$

- $\phi(n) = (p-1)(q-1) = 60$

- Privát kulcs  $(n, d)$ :

- $d = 43$ , amelyre

- $e \cdot d = 1 \bmod \phi(n)$

- Visszakódolás:

- $\text{message} = \text{code}^{43} \bmod 77$

$47^{43} \bmod 77$

$$= 47^{1+2+8+32} \bmod 77$$

$$= 47 \cdot 47^2 \cdot 47^8 \cdot 47^{32} \bmod 77$$

$$= 47 \cdot 47^2 \cdot ((47^2)^2)^2 \cdot (((47^2)^2)^2)^2 \bmod 77$$

= ...

$$= 5 = \text{message}$$

- Nyilvános kulcs  $(n, e)$ :

- $n = 77$  és egy szám  $e = 7$

- Titkosítás:

- $\text{code} = \text{message}^7 \bmod 77$

- $\text{message} = 5$

- $\text{code} = 5^7 \bmod 77 = 47$

## Elektronikus aláírás

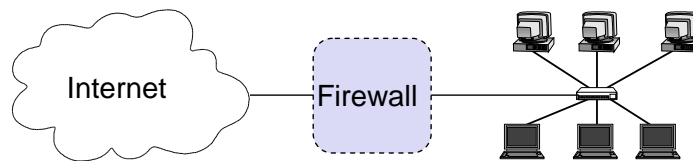
- Más néven digitális szignatúra
  - Az aláírónak van egy (titkos) privát kulcsa
  - A dokumentumot a privát kulcs felhasználásával írja alá
  - és a nyilvános kulccsal verifikálható, hogy az aláírás tőle származik
  - A nyilvános kulcs mindenki számára ismert
- Példa egy aláírás sémára
  - `text`: üzenet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

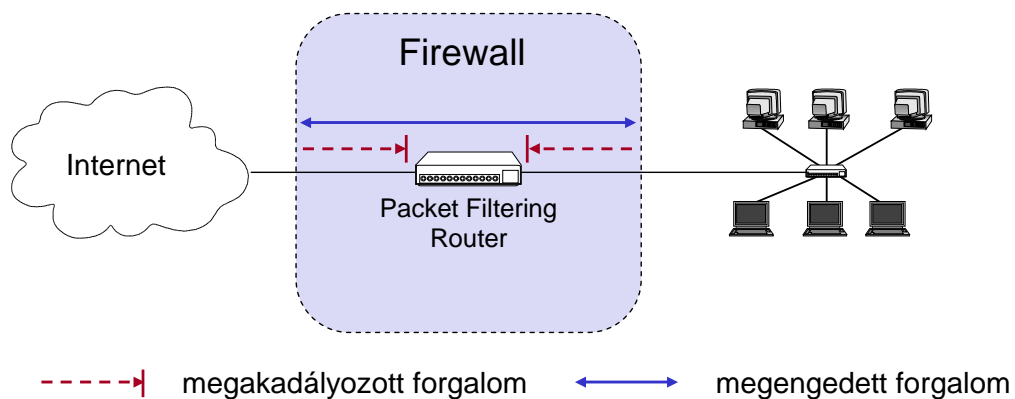
- NAT (Network Address Translation)
- Basic NAT (Static NAT)
  - Minden belső IP cím egy külsővel helyettesítődik
- Hiding NAT = PAT (Port Address Translation) = NAPT (Network Address Port Translation)
  - A socket-pár (IP-cím és Port-szám) átszámítódik
- Módszerek
  - A különböző lokális számítógépeket 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

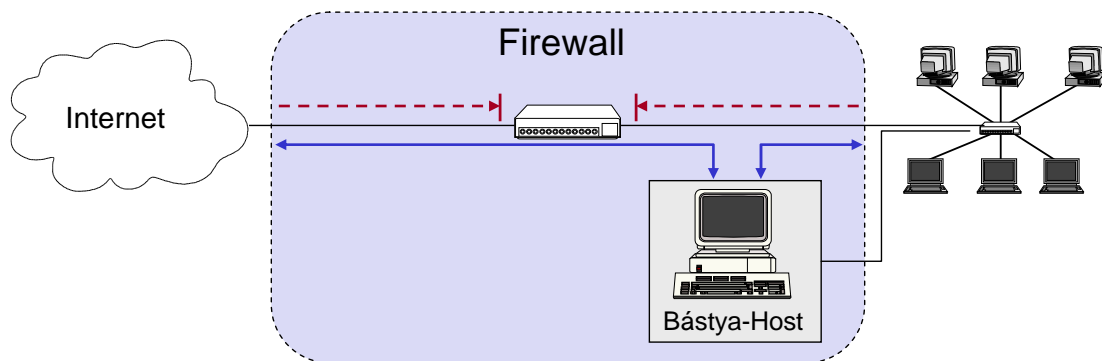
## Tűzfal architektúra – egyszerű csomagszűrő

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



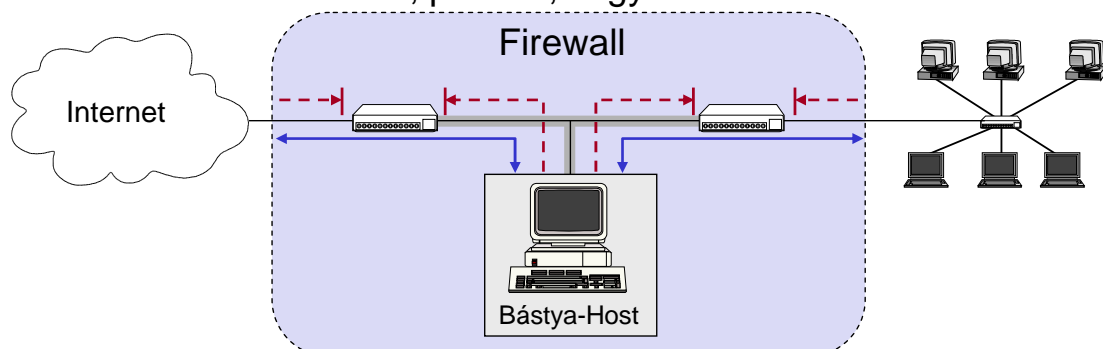
## Tűzfal architektúra -- Screened Host

- 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