

# EA10

**Határidő** Nincs megadva határidő**Pont** 10**Kérdések** 10**Elérhető** nov 29, 08:30 - nov 29, 09:15 körülbelül 1 óra**Időkorlát** 15 perc

## Próbálkozások naplója

	Próbálkozás	Idő	Eredmény
LEGUTOLSÓ	<a href="#">1. próbálkozás</a>	7 perc	8 az összesen elérhető 10 pontból

 A helyes válaszok el vannak rejtve.

Ezen kvíz eredménye: **8** az összesen elérhető 10 pontból

Beadva ekkor: nov 29, 09:03

Ez a próbálkozás ennyi időt vett igénybe: 7 perc

### 1. kérdés

**1 / 1 pont**

Mely állítás igaz a TCP lassú indulás (slow start) mechanizmusára?



A torlódási ablakot egy a csomagvesztés óta eltelt idővel paraméterezett köbös függvény segítségével határozzuk meg.



Minden nyugta fogadása esetén a küldő egy szegmenssel növeli a torlódási ablakot. Az időben ez gyors, exponenciális növekedést jelent a küldési rátában.



Minden beérkezett nyugta esetén "1/torlódási ablak"-kal növeljük a torlódási ablakot. Ez időben lineáris növekedést eredményez.



Minden nyugta fogadása esetén a küldő egy szegmenssel növeli a torlódási ablakot. Az időben ez lassú, lineáris növekedést jelent a küldési rátában.

## Helytelen

## 2. kérdés

0 / 1 pont

Hogyan állítjuk be az újraküldéshez használt időkorlátot (RTO) a TCP esetén?

- ☐  $2 * RTT$
- ☐  $\alpha * RTO_{old} + (1-\alpha)*RTT$
- ☒ RTT
- ☐ 2

## 3. kérdés

1 / 1 pont

Igaz-e az állítás: TCP SYN flood támadás azt használja ki, hogy egy szerver minden beérkező SYN csomaghoz erőforrást foglal a kapcsolat állapotának nyilvántartásához, mely akár 2 percig is lefoglalva maradhat. Amikor rövid időn belül sok ilyen csomag érkezik a rendelkezésre álló erőforrások elfogynak és a normális kapcsolatok visszautasításra kerülnek/extrém esetben a szerver összeomlik.

- ☒ Igaz
- ☐ Hamis

## 4. kérdés

1 / 1 pont

Mely állítások igazak az TCP AIMD mechanizmusára? (Több válasz is lehet helyes.)

- ☐ Gyors újraküldés esetén csomagvesztés során (dupack) egy szegmenssel csökkentjük a torlódási ablakot.



Minden nyugta fogadása esetén "1/torlódási ablak méret"-tel növeljük torlódási ablakot.



Gyors újraküldés esetén csomagvesztés során (dupack) felére csökkentjük a torlódási ablakot.



Minden nyugta fogadása esetén egy szegmenssel növeljük a torlódási ablak méretét.

### 5. kérdés

1 / 1 pont

Igaz-e az állítás: Tipikus webes forgalom esetén a TCP hatékony, képes kihasználni a rendelkezésre álló szabad hálózati kapacitást (sávszélességet).

☐ Igaz

☒ Hamis

### 6. kérdés

1 / 1 pont

Mit nevezünk torlódásnak?

☐ A hálózat szírt pontja nulla.

☒ A hálózat terhelése nagyobb, mint a kapacitása.

☐ A hálózat kapacitása nagyobb, mint a terhelése.

☐ A hálózat könyök pontja nagyobb, mint a kapacitása.

**7. kérdés****1 / 1 pont**

Mit jelent a 3 nyugta duplikátum fogadása a TCP RENO esetén?

- ☐ Az ablakot 3 szegmenssel csúsztatja a küldő.
- ☒ Csomagvesztést jelez.
- ☐ Olyan, mintha egy nyugta érkezett volna.
- ☐ Helyes átvitelt jelez.

**Helytelen****8. kérdés****0 / 1 pont**

Mit csinál Nagle algorimusa a TCP esetén, ha van nem nyugtázott adat és az elérhető adat < MSS

- ☐ Várakoztatja az adatot egy pufferben, amíg nyugtát nem kap.
- ☐ Vár egy időegységet és elküldi az adatot.
- ☒ Eldobja az adatot.
- ☐ Elküldi az adatot

**9. kérdés****1 / 1 pont**

Igaz-e az állítás: Csomagvesztés utáni kis időben a CUBIC TCP küldési rátájának a felfutása gyorsabb, mint amit a slow start mechanizmus esetén láttunk.

- ☒ Igaz
- ☐ Hamis

**10. kérdés****1 / 1 pont**

Mely állítások igazak a Compound TCP-re? (Több helyes válasz is lehet.)

☐ A torlódási ablakot egy köbös függvény segítségével határozza meg.

☒ Csomagvesztés alapú torlódási ablakot is fenntart.

☐ A csomagvesztést nem tekinti a torlódás jelének.

☒ Késleltetés alapú torlódási ablakot is fenntart.

☒ Reno alapú

Kvízeredmény: **8** az összesen elérhető 10 pontból