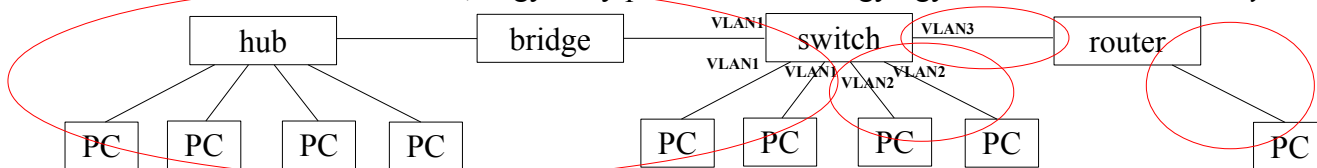


Vizsga feladatok számítógép-hálózatok tárgyból

Minden kérdésnél 1 pont szerezhető, összetett kérdéseknél részpont is kapható. Nem működő UNIX parancs nem ér pontot. Az elégséges osztályzathoz legalább a pontok 60%-át, azaz 9 pontot kell megszerezni.

1. Ön rendszergazdaként dolgozik Linux alatt. Állítsa be, hogy a `/tmp/leiras` nevű könyvtár tulajdonosa a `doku` nevű felhasználó, csoporttulajdonosa pedig a `ment` nevű csoport legyen!
`chown doku:ment /tmp/leiras` (esetleg `chown doku.ment /tmp/leiras`) vagy
`chown doku /tmp/leiras; chgrp ment /tmp/leiras`
2. Állítsa be, hogy a `/tmp/leiras` nevű könyvtárra a tulajdonosnak minden joga meglegyen, a csoporttársak tudják listázni és elérjék a benne levő fájlokat, de ezenkívül senkinek semmilyen más joga ne legyen rá!
`chmod 750 /tmp/leiras`
3. Hogyan választaná meg a `p` paraméter értékét *p-perzisztens CSMA* esetén, ha a forgalmi igény várhatóan
 - a) a csatornakapacitás 10%-a körüli `p=1`
 - b) a csatornakapacitás többszöröse `p=0` vagy `p=0.1` (A két érték között minden jó, a 0.1 nem szigorú korlát.)
4. Ethernetnél mit jelent a *promiscuous mode*?
A hálózati interfész válogatás nélkül minden keretet vesz.
5. Mit tud a 100BaseTX Ethernetről?
100Mbit/s adatsebesség, Cat5 kábel szükséges, csillag topológia, max 100 m szegmenshossz, full duplex módra képes, ekkor egy érpáron ad, és egy érpáron vesz, működhet half duplex módban is, 4b/5b kódolás miatt 125Mbaud jelzési sebesség, stb.
Pontozás: minden lényegi információ 0.2 pont, jelentéktelenek 0.1 pont; összesen max. 1 pont adható.
6. Az alábbi vázlaton karikázza be, hogy mely portok tartoznak egy-egy *üzenetszórás*i tartományba!



7. Vonja össze a lehető legnagyobb mértékben a következő hálózatokat: 10.1.1.0/26, 10.1.1.64/26, 10.1.1.128/25, 10.1.2.0/24, 10.1.4.0/23
 10.1.1.0/26: 10.1.1.0|000000 +
 10.1.1.64/26: 10.1.1.01|000000 => 10.1.1.0|0000000, azaz: 10.1.1.0/25, folytatva:
 10.1.1.0/25: 10.1.1.0|00000000 +
 10.1.1.128/25: 10.1.1.1|00000000 => 10.1.1.|000000000, azaz: 10.1.1.0/24, de tovább nem vonható össze.
 Végeredmény: 10.1.1.0/24, 10.1.2.0/24, 10.1.4.0/23
8. A 201.206.12.0/23 hálózatot bontsa fel a lehető legtöbb olyan hálózatra, amelyekre legalább 50 számítógépet rá lehet kötni!
A gépcím résznek 6 bit kell, így a maszk /26 lesz.
201.206.0000110|0.00|000000 – 3 biten az összes lehetséges esetet leírva kapjuk:
201.206.12.0/26, 201.206.12.64/26, 201.206.12.128/26, 201.206.12.192/26,
201.206.13.0/26, 201.206.13.64/26, 201.206.13.128/26, 201.206.13.192/26.
9. A 10.1.4.0/22 hálózatban a router a legnagyobb kiosztható IP címet kapta. A router látja el a névkiszolgáló feladatát is. Linux alatt állítsa be hálózat működéséhez szükséges paramétereket a 10.1.4.1 IP című gépen!
`ifconfig eth0 10.1.4.1 netmask 255.255.252.0 broadcast 10.1.7.255 up`
`route add default gw 10.1.7.254`
`echo "nameserver 10.1.7.254" > /etc/resolv.conf`

10. Mutassa be az AIMD algoritmus TCP-ben használt változata működésének lényegét! Miért célszerű ez?

Minden RTT alatt: (0.1 pont)

- Növeljük congestion window értékét MSS-sel, ha nincs torlódásra utaló jel. (0.3 pont)

- Csökkentsük congestion window értékét a felére, ha van torlódásra utaló jel. (0.3 pont)

Az algoritmus használata azért célszerű, mert a növelés csak lineáris (azaz viszonylag lassú), a csökkentés viszont exponenciális; torlódás esetén nagyon gyorsan a felére, negyedére, stb. tud csökkenni a congestion window. (0.3 pont)

Ahol:

- RTT: Round-Trip Time: az az időtartam, ami TCP szegmens elküldésétől a szegmens nyugtázására alkalmas nyugta megérkezéséig eltelik. (0.2 pont)

- MSS: Maximum Segment Size: a TCP szegmens számára megengedett maximális méret. (0.2 pont)

De összesen csak max. 1 pont adható a feladatra!!!

(A pontozás indoklása: a magyarázatok fontosak, nélkülük nem igazán definiált a működés, de ha a hallgató ezeket *ismertnek tételezte fel*, akkor is elérheti az 1 pontot.)

A *slow start* nem része a válasznak. Ha odaírják nem tekintjük hibának, de pont nem jár érte.

11. Miért káros, ha *ARP Probe* helyett *Gratuitous ARP* üzeneteket használnak? (Két dolgot említsen.)

- Nem óvja meg a már működő gépek működőképességét.

- Nem teszi lehetővé a most induló gépnél sem azt, hogy automatikusan (emberi beavatkozás nélkül) más IP-címet használjon.

12. Sorolja fel az IPv6 *állapotmentes automatikus címkonfigurációja* során használható ICMPv6 üzeneteket, és azt, hogy mikor, mire használják őket!

Negihbor Solicitation: link-lokális cím, illetve globális unicast cím előállítása után azok egyediségének ellenőrzésére használják. (0.5 pont)

Neighbor Advertisement: a fentiek esetén, azokra válaszként fordulhatna elő, de nem valószínű, hogy fog! (0.3 pont)

Router Solicitation: hálózati prefix kérése (a globális unicast cím előállításához kell) (0.5 pont)

Router Advertisement: hálózati prefixet hirdeti a router (a fentire válaszként vagy anélkül is) (0.5 pont)

De összesen csak max. 1 pont adható a feladatra!!!

13. Mít tud az IPv6 szabványos dokumentációs prefixéről?

Annak érdekében, hogy a dokumentációkban szereplő példák ne okozzanak zavart (a fejekben) vagy működő rendszerekkel való ütközést, lefoglaltak egy globális unicast prefixet kifejezetten dokumentációs célra. Ez a 2001:DB8::/32 (RFC 3849). Ezt a prefixet soha senkinek sem fogják kiosztani és nem is routolják. (De természetesen ettől még nem számít lokális unicast prefixnek!)

Pontozás: a prefixre csak akkor adható pont, ha helyes, ekkor 0.5 pontot ér. Ha a magyarázatból kiderül, hogy mire jó, akkor a magyarázat 0.5 pontot ér, ha nem, akkor a szöveg 0 pontot ér.

14. DNS-nél mi a különbség a *recursive* és az *iterative query* között? (Melyikkel mit kell tenni?)

Ha egy névkiszolgáló *recursive query* kap, akkor arra teljes választ kell adnia (IP cím vagy hibajelzés), ehhez szükség esetén *iterative query* sorozatát kell végrehajtania.

Ha egy névkiszolgáló *iterative query* kap, akkor arra elegendő egy *referral*al válaszolnia.

Ennyit várunk el, bár jóval többet el lehetne mondani...

15. Ismertesse az eseményvezérelt diszkrét idejű szimuláció algoritmusát!

Inicializálás, azaz bizonyos események felidőztítése a FES-be;

REPEAT

Legkisebb időbélyegű esemény kivétele (és törlése) a FES-ből;

MOST := a kivett esemény időbélyege

Esemény feldolgozása, eközben esetleg újabb esemény(ek) generálása (és behelyezése a FES-be);

UNTIL (elfogytak az események) v (MOST > mint egy beolvasott határ) v (egyéb ok miatt meg kell állni)