**Experimenty s bandwidth, priority a WFQ pre nastavenie QoS**

**Topológia:**

PC1 (Wireshark +I/O graphs,statistics, D-ITG+logger) --- R1 --- R2 --- PC2 (Wireshark +I/O graphs,statistics, D-ITG+logger)

Poznámka: Ak sa bije telnetové pripojenie z VM Debian s pripojením telnet vo Windowse cez PuTTy, treba zadať vo VM Debian > Machine > Stop.

* Pre sériovú linku nastavte clock rate aj bandwidth na 128 kbps
  + Pozor, BW je potrebné, pretože podľa neho sa alokuje veľkosť HW buffra. Service policy s touto hodnotou potom porovnána súčet požiadaviek na prenos daným rozhraním, podobne aj smerovacie protokoly túto hodnotu používajú pri výpočtoch.
* Generujte 3 toky prevádzky (generovania sa týkajú iba čierne záznamy v tabuľke):

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tok | Intenzita toku | Veĺkosť paketov | Rozdelenie medzier medzi paketmi | L4 | Cieľový port | Trvanie generov. | Začiatok generovania | Ako to označkovať na vstupe |
| Tok 1 | 90 kb/s  (BW 50) | 512 B/p. | konšt.  intervaly | UDP | 9001 | 90 s. | 0 | AF11 |
| Tok 2 | 200  (BW 20) | 512 B/p. |  |  | 9002 | 90 s. | 30 sek. | AF21 |
| Tok 2 | 50  (BW 20) | 64 B/p. |  |  | 9003 | 90 s. | 60 sek. | AF31 |

* + Údaje pre intenzitu toku si prepočítajte na pakety za sek pre použitie v DITG
  + Pokúste sa pripraviť 2 scenáre (resp. 3) pre každú jednu úlohu (z troch ktoré nasledujú)
    - Celkový tok pod kapacitou linky
    - Celkový tok prevyšujúci kapacitu linky
  + ACL 101 - povoľ iba pakety z prvého toku.. port 9001
  + ACL 102 - povoľ iba pakety z druhého toku.. port 9002
  + ACL 103 - povoľ iba pakety z tretieho toku.. port 9003
* Definuj 3 triedy prevádzky pomocou class-map (TokX je názov triedy) a match na vytvorený ACL
  + Tok1 – match access-group 101
  + Tok2 – match … 102
  + Tok3 – match … 103
* Nastav značkovanie paketov na vstupe do svojho smerovača z vašej LAN pomocou policy-map
  + Všetko robí každý na svojom smerovači
  + Názov policy-map: **ZnackovanieNaVstupe**
  + Pre Tok1 daj značku DSCP=AF11
  + Pre Tok2 daj značku DSCP=AF21
  + Pre Tok3 daj značku DSCP=AF31
* Vygeneruj dané dáta a over či funguje ako sa očakáva
  + Over značkovanie (spolupráca dvojice, aby ste otestovali značkovanie jedným aj druhým smerom)
  + Zaznamenaj si grafy ako vyzerajú jednotlivé toky na strane prijímača (IO graphs Wireshark)
    - Dobre viditeľné výsledky by mohli byť pri týchto nastaveniach:
      * X axis: tick interval 1sec, pixels per tick: 5
      * Y axis: bits/tick, scale 100000, **Smooth** M.average: 4
      * Ak sa vám osvedčilo iné nastavenie, uveďte aké
* Príkazy na overenie:
  + Show policy-map int s0/0/0
  + Show policy-map int s0/0/0 out
  + Show policy-map intf0/0 in



**Úloha 1: Obmedzenie tokov pomocou bandwidth (BW)**

* Nastavte bandwidth v pomere 50:20:20 pre jednotlivé 3 toky
* Dali by sa použiť už vytvorené triedy Tok1, Tok2, Tok3, ale... aby sme sa priblížili tomu, ako sa to deje v praxi (obmedzenie sa nemusí diať na tom istom smerovači na ktorom sa značkuje), vytvoríme si 3 nové triedy s názvom TriedaAF11, TriedaAF11, TriedaAF11, ktoré následne využijeme v QoS politiky v ktorej ich obmedzíme pomocou BW, t.j. postupne:
* Definuj 3 triedy prevádzky pomocou class-map (TriedaAFXY je názov triedy) a match podľa DSCP
  + TriedaAF11– match dscp AF11
  + TriedaAF11– match dscp AF21
  + TriedaAF11– match dscp AF31
* Obmedz prevádzku z daných troch tried v pomere 50:20:20 pomocou policy-map a príkazu BW
  + Názov policy-map: **ObmedzenieNaVystupeCezBW**
  + Pre triedu TriedaAF11 nastav BW 50
  + Pre triedu TriedaAF21 nastav BW 20
  + Pre triedu TriedaAF31 nastav BW 20
  + Pozor túto QoS politiku treba nasadiť pri výstupe z daného smerovača, t.j. na sériové rozhranie smerom k susednému smerovaču (spravia to obaja v dvojici, aby si každý vyskúšal všetko, testovať to treba potom jednotlivo, najprv jeden je receiver, potom druhý je receiver..)
* Generuj dáta a odsleduj výsledky viditeľné v IO graphs vo Wiresharku
  + Vo Wiresharku viete použiť filter podľa DSCP:
    - Reciever – Statistics – IO graph: AFxy = 8 \* x + 2 \* y  
       ip.dsfield.dscp == 10 AF11 8 x 1 + 2 x 1 = 10   
       ip.dsfield.dscp == 18 AF21 8 x2 + 2 x 1 = 18  
       ip.dsfield.dscp == 26 AF21 8 x3 + 2 x 1 = 26
  + Dajte si pozor, aby ste mali správnu "mierku/zoom" pri pohľade na zobrazované dáta v IO graphs
  + Ako sa líšia výsledky (porovnaj grafy) oproti tomu keď sme nemali nasadenú túto QoS politiku?

**Úloha 2: Ako sa dá riešiť problém zahltenej linky výberom správnej frontovacej disciplíny - WFQ**

* Nemažte pôvodné politiky, iba ich budeme postupne odoberať z daných rozhraní
* Odober politiku **ObmedzenieNaVystupeCezBW**
* Nastav novú QoS politiku s názvom **RieseniePomocouWFQ**
  + Vytvor túto politiku, ktorá pre triedu class-default nastaví systém vyberania z front na režim WFQ (príkaz fair-queue)
  + Nasaď túto potiku na sériové rozhranie pre pakety odchádzajúce z tohto rohrania
* Opakuj generovanie 3 tokov pomocou DITG a odsleduj výsledky cez IO graphs
  + Tu si zaznamenajte do reportu na akej verzii IOSu pracujete, pretože:
    - IOS 12.4 pri výpočte hash kódu berie do úvahy hodnoty DSCP (naše AF ktoré sme ešte nezrušili) a pásmo rozdelí v pomere 90:200:50 (to sú naše intenzity tokov, ktoré generujeme DITG)
      * Zistite či toto platí aj pre váš prípad, nejaký dôkaz doložte do vášho reportu
      * V tomto prípade spravte aj experiment, že zrušíte politiku, ktorá vám značkuje pakety pri vstupe na váš smerovač, a odsledujete potom výsledky politiky RieseniePomocouWFQ - pokúste sa zistiť či vidieť rozdiel, ako sa delí pásmo v jednom a druhom prípade (s ponechaním značkovania a bez neho)
    - IOS 15.1 pri výpočte neberie do úvahy hodnotu DSCP
      * Zistite v tomto prípade, ako rozdelí prevádzku a v akom pomere
      * Experimentálne odoberte značkovaciu politiku zo vstupného rozhrania na vašom smerovači a spravte opätovné meranie výsledkov, ako sa delí prevádzka, v akom pomere?

**Úloha 3: Obmedzenie tokov pomocou priority**

* Podobne ako v predošlej úlohe chceme deliť kapacitu danej linky v pomere 50:20:20, ale pre prvý tok, chceme vytvoriť prioritnú frontu s pridelením 50% kapacity linky
  + Príkaz "priority 50" nastavuje tzv. implicitnú prioritu, t.j. obmedzenie tohto toku začne fungovať až vtedy, keď súčet všetkých tokov začne presahovať kapacitu linky, t.j. začne fungovať až vtedy, keď je dané rozhranie zahltené/preťažené. V takejto situácií (linka je preťažená) bude znižovať pásmo pre tento tok, začne ho uvoľňovať pre ostatné toky, až sa zastaví na 50kbps.
  + Príkaz "police 50" (budeme mať až neskôr) sa nastavuje explicitná priorita, kedy toku pridelí 50kbps vždy, t.j. aj vtedy keď je ešte voľná kapacita linky.
* Znova nasaďte značkovaciu politiku na vstupe na váš smerovač.
* Nasaďte novú politiku s názvom **ObmedzenieNaVystupeCezPriorityAjBW**
  + **Scenár A.:** Politika bude podobná ako sme mali ObmedzenieNaVystupeCezBW, zmena bude iba pre triedu TriedaAF11, pre ktorú treba nastaviť namiesto bandwidth 50 (no bandwidth 50), zadať priority 50
    - Odsledujte a zaznamenajte výsledky cez IO graphs, porovnajte s výsledkami predošlých 2 úloh
  + **Scenár B.:** Zmeňte aj triedu TriedaAF21 bandwidth na priority (no bandwidth 20, priority 20)
    - Odsledujte a zaznamenajte výsledky cez IO graphs, porovnajte s výsledkami predošlých úloh
  + **Scenár C.:** Zmeňte aj triedu TriedaAF31 bandwidth na priority (no bandwidth 20, priority 20)

**Poznámka na záver:** Kto ešte nemá jasno a chce sa zorientovať v príkazoch bandwidth a priority:

<http://www.cisco.com/c/en/us/support/docs/quality-of-service-qos/qos-packet-marking/10100-priorityvsbw.html#whichtrafficclassescanuseexcessbandwidth>