# Policing a shaping

Pozrieme z Moodla: **Prednáška 4**.pptx – slajd **13 až 32**.

**R1**

**R2**

**10.0.1.0/24**

**10.0.2.0/24**

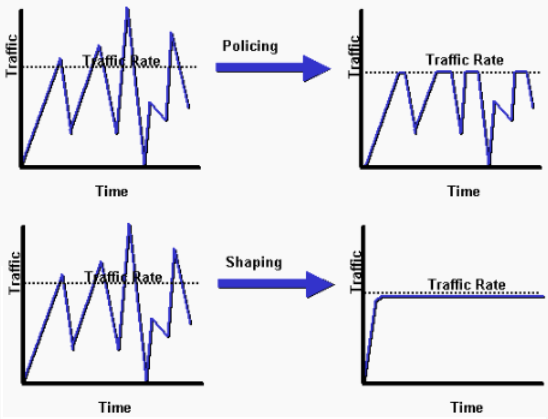
**10.0.12.0/24**

**128 k**

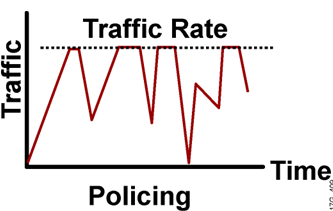
**Sender**

**Reciever, WireShark**

**Obmedzovanie (policing) vs. tvarovanie (shaping) prevádzky**

Diagram ilustruje základný rozdiel. Keď sa rýchlosť sieťovej prevádzky priblíži maximálnej rýchlosti, ktorá je nakonfigurovaná na danom rozhraní smerovača, časť prevádzky, ktorá je nad rámec povoleného maxima sa zahodí (alebo označkuje). Výsledkom je „zubaté“ orezanie prevádzky, ktoré ilustruje obrázok vľavo. Naproti tomu pri tvarovaní prevádzky si smerovač tú časť prevádzky, ktorá presahuje povolenú maximálnu rýchlosť, odloží a pokúsi sa preniesť neskôr. Výsledkom je vyhladená krivka rýchlosti paketov v čase. Pri tvarovaní prevádzky je nutná existencia odkladacieho frontu, a dostatočné množstvo pamäte na odloženie paketov, ktorých prenos sa týmto oneskorí. Pri obmedzovaní prevádzky takýto front nemusí byť prítomný vôbec. Ďalším rozdielom je, že obmedzovať prevádzku možno na vstupnom alebo výstupnom rozhraní smerovača, kým tvarovanie prevádzky možno použiť len na výstupnom fronte. Pri tvarovaní prevádzky je potrebný mechanizmus pre plánovanie paketov na ich neskoršie prenesenie, ktorým môže byť napr. CBWFQ alebo LLQ, čiže buď mám jeden odkladací front, alebo viacero.  
<http://www.cisco.com/c/en/us/support/docs/quality-of-service-qos/qos-policing/19645-policevsshape.html>

**Policing (tým začneme)**

****

* obmedzuje prevádzku (aj keď rozhranie nie je preplnené)
* môžeme aj vo vstupnom aj výstupnom smere
  + zvyčajne ale na vstupe
* vedro s tokenmi sa dopĺňa priebežne, token = povolenie vyslať 1 bit

**Budeme sledovať ako sa občas prejaví „rezerva“ v policingoch:**

* 1-úrovňový 2- farebný (základná verzia jednoduchá)
* 1-úrovňový 3- farebný (srTCM = single rate three color marking)
* 2-úrovňový 3-farebný (trTCM = two rate three color marking)

**Generované dáta (D-ITG):**

* Tok: kolísavý okolo: a.) aj b.) 84 kb/s UDP, c.) 112kbps (kolísavý chceme preto, aby sme videli obmedzovanie a tvarovanie/vyhladzovanie prevádzky)
  + Intenzita: 35 pak/s - náhodné časy príchodov s expon. rozdelením, pričom priemerne bude chodiť 35 pak/s
  + Veľkosť paketov náhodná - spravte pre každý scenár (viď nižšie) tieto 3 varianty:
    - a.) s rovnomerným rozdelením = uniform od 100 do 500 B
    - b.) s exponenciálnym rozdelením so strednou veľkosťou paketu 300 B
    - c.) s exp. rozdelením so strednou dĺžkou 400 B
* Policing: 84 kb/s

**Scenáre pre experimenty s policingom:**

1. **Single rate two color (jednoduchý token bucket)**
   1. Obmedzovať prevádzku (policing) možno tak pre triedu class-default ako aj pre ktorúkoľvek špecifickú triedu (my budeme robiť tú druhú možnosť). Obmedzovať budeme čo najbližšie k zdroju, takže na vstupnom porte nášho smerovača z danej LAN:
      1. Vytvorte triedu „ZAKAZNIK“ (class-map), do ktorej spadne všetka generovaná UDP prevádzka D-ITG generátorom (najjednoduchšie je použiť vhodný ACL)
      2. Vytvorte politiku „POLICING1\_DROP“ (policy-map) pre obmedzovanie (policing) prevádzky nasledovne:
         1. Všetky pakety triedy ZAKAZNIK nech označkuje DSCP hodnotou 2
         2. Pakety nad dovolený rámec 84 kbps (CIR) nech zahadzuje, ostatné nech prepúšťa.
      3. Politiku aplikujte na f0/0 vo vstupnom smere.
      4. Vygenerujte prevádzku zo Sendera na Receiver v D-ITG
      5. Využite výpis *sh policy-map int f0/0* a zistite:
         1. akú veľkosť zhluku (burst) prepúšťa váš smerovač
            1. Hľadajte Bc = veľkosť vedra = max. počet tokenov vo vedre

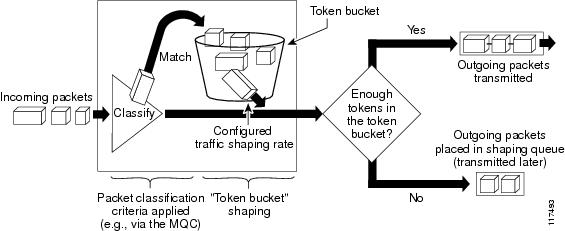
Zistený údaj prepočítajte na bity

* + - 1. koľko krát za sekundu sa dopĺňa vedro tokenov (1/Tc):
         1. vypočítaj interval dopĺňania Tc = Bc / CIR [sek.]
         2. potom 1/Tc = …… [doplnení/sek.]
    1. Preskúmajte IO graphs vo Wiresharku, spravte screen shots do vášho reportu
  1. Teraz pakety presahujúce limit nezahadzujte, ale označkujte ich DSCP značkou af12, aby sme ich videli v IO graphs vo Wiresharku:
     1. Vytvorte politiku „POLICING1\_FARBI“
        1. Všetko bude ako predošlé, akurát *exceed-action* sa zmení z *drop* na *set-dscp-transmit af12*
        2. Najprv deaktivujte predošlú politiku, následne aplikujte novú na to isté rozhranie.
     2. Preskúmajte IO graphs vo Wiresharku, spravte screen shots do vášho reportu
  2. Preskúmajte, akú najmenšiu hodnotu CIR v príkaze police váš smerovač povoľuje
     1. Zistite ako sa potom zmení Bc?
     2. Nikdy sa však neodporúča ísť pod veľkosť paketu, aby za čas medzi paketmi stihli natiecť tokeny aspoň na jeden paket.

1. **Single Rate Three Color Marker (srTCM)**
   1. Pritekajúce tokeny, ktoré sa nezmestia do prvého vedra, ktorého veľkosť je Bc = comform burst, sa budú ukladať do druhého vedra veľkosti Be = excess burst
      1. Vytvorte novú politiku POLICING2\_srTCM:
         1. Všetky pakety triedy ZAKAZNIK nech označkuje DSCP hodnotou 2
         2. CIR ostáva ako v predošlom 84 kbps
         3. Pridáme ale nastavenie pre Bc = 3000, Be = 3000
         4. Pakety, pre ktoré nie je dostatok tokenov v prvom vedre, ale je dostatok tokenov v druhom vedre, nech zafarbí/označkuje DSCP hodnotou af12 (exceed-action...)
         5. A pakety, pre ktoré nie je dostatok ani v prvom ani v druhom vedre, nech označkuje af13 (violate-action...)
         6. Najprv deaktivujte predošlú politiku, následne aplikujte túto novú na to isté rozhranie.
      2. Preskúmajte výstupy IO graphs vo Wiresharku aj sh policy... a popíšte akú zmenu vidíte.
2. **Two Rate Three Color Marker (trTCM)**
   1. Teraz chceme obmedziť tok na maximálnu ustálenú úroveň PIR=84kbps, ale pritom identifikovať jeho časť na ustálenej úrovni CIR=60kbps (**C**IR – ustálená rýchlosť na konformnej úrovni = **c**onform = rýchlosť akou sa bude plniť prvé vedro, **P**IR – ustálená rýchlosť na maximálnej povolenej úrovni = **p**eak = rýchlosť akou sa bude plniť druhé vedro). Každé vedro sa plní svojou rýchlosťou. Odoberanie tokenov pre paket, ktorý príde vo chvíli, že sa preň nájde dostatok tokenov v jednom aj druhom vedre, sa deje z jedného aj druhého vedra a paket sa prepošle.
      1. Vytvorte novú politiku POLICING3\_trTCM:
         1. Všetky pakety triedy ZAKAZNIK nech označkuje DSCP hodnotou 2 (ako v predošlom)
         2. Tu ale chceme zmenu pre CIR = 60kbps a PIR = 84 kbps
         3. Akcie ostanú ako v predošlom scenári:
            1. Pakety, pre ktoré nie je dostatok tokenov v prvom vedre, ale je dostatok tokenov v druhom vedre, nech zafarbí/označkuje DSCP hodnotou af12 (exceed-action...)
            2. A pakety, pre ktoré nie je dostatok ani v prvom ani v druhom vedre, nech označkuje af13 (violate-action...)
         4. Najprv deaktivujte predošlú politiku, následne aplikujte túto novú na to isté rozhranie.
      2. Preskúmajte výstupy IO graphs vo Wiresharku aj sh policy... a popíšte akú zmenu vidíte.
      3. Akú zmenu by sme videli, keby sme zvolili CIR=84kbps a PIR=120kbps?
   2. Vypočítajte čas Tc z výpisu CIR a PIR zo show policy-map int f0/0, či je rovnaký interval doplnenia tokenov:
      1. Tc = Bc / CIR = .... sek.
      2. Tc = Be / PIR = .... sek.

**Shaping**

* jednoduchá konfigurácia v porovnaní s policing-om, typicky sa dáva na výstupné rozhranie



* implementovaný pomocou jednoduchého token bucketu, veľkosti Bc+Be
  + ale !!! tento token bucket sa dopĺňa len diskrétne, v intervaloch Tc (v policingu sa dopĺňal spojito)
    - **shape average** doplní Bc tokenov každých Tc sekúnd (1 token je aj tu stále 1 bit)
    - **shape** **peak** doplní Bc+Be tokenov každých Tc sekúnd
      * **shape peak** <CIR> <Bc> <Be>
        + the actual maximum sending rate is limited to:  
          PIR = CIR\*(1+Be/Bc).
        + That is, each time interval Tc=Bc/CIR the shaper allows sending up to Bc+Be bits of data. By default, if you omit the value for Be, it equals to Bc and thus PIR=2\*CIR by default. However, due to some IOS show output discrepancy, this is NOT reflected in “show” command output, unless you explicitly specify the Be value in command line.
        + With shape peak configured this way, you can see both CIR as the “**average rate**” and PIR as the “**target rate**” when issuing “show policy-map” command. – toto by mala byť výhoda, že presnejší výpis.
        + <http://blog.ine.com/2008/08/26/understanding-the-shape-peak-command/>

**Scenáre pre experimenty so shapingom:**

* preskúmať sh policy-map... a IO graphs
  + Zistite v čom je rozdiel vo výpisoch pre nasledovné dva scenáre, môžete spraviť aj viac rôznych, s rôznymi nastaveniami, ak je potrebné, aby bolo vidieť rozdiel

1. **Nastavte tvarovanie prevádzky s ohľadom na priemer (shape average)**
   1. Podobne ako pri policingu, využite triedu ZAKAZNIK, a vytvorte policy-map SHAPING\_AVERAGE na CIR = 84 kbps
2. **Nastavte tvarovanie prevádzky s ohľadom na špičku (shape peak)**
   1. Podobne ako pri policingu, využite triedu ZAKAZNIK, a vytvorte policy-map SHAPING\_PEAK na CIR = 84 kbps