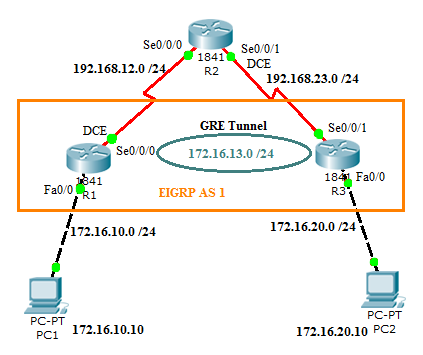
**Lab: GRE tunel, QoS pre-classify**  
*(Predáška 5 - QoS vo VPN sieťach***)**

**Graf topológie**

* Linku medzi R2 a R3 zmeníme na ethernetovú a pridáme tam prepínač, kvôli port mirroring tunolovacej prevádzky



**Ciele**

* Nakonfigurujte GRE tunel
* Nakonfigurujte QoS pre-classify
* Overte funkčnosť pre-classify operácií

**Scenár**

Budeme implementovať vlastnosti "QoS pre-classify", aby nástroje založené na kontrole príslušnosti k tokom (flow-based) mohli robiť inteligentnejšie rozhodnutia pri poskytovaní šírky pásma pre toky nachádzajúce sa v GRE tunely.

GRE tunel (Generic Routing Encapsulation), aj IPsec tunel kopírujú v priebehu enkapsulácie pre každý paket obsah poľa ToS/DiffServ z vnútornej IP hlavičky a umiestňujú ho do vonkajšej hlavičky. Všetky nástroje založené na kontrole príslušnosti k toku (flow-based) budú vďaka tomu schopné vidieť IP prioritu alebo DSCP značku po celej trase tunela.

Weighted fair queuing (WFQ) povoľuje smerovaču radiť pakety do frontov podľa príslušnosti k toku alebo konverzácií, do ktorej paket patrí. Samotný tok je definovaný podľa zdrojovej a cieľovej IP adresy, čísla portu, použitého transportného protokolu a hodnoty IP priority. WFQ potom šírku pásma spravodlivo rozdeľuje medzi všetky toky. Keďže naša prevádzka bude uzavretá v tunely, na klasifikáciu bude možné použiť iba ToS/DiffServ byte (kopíruje sa do vonkajšej tunelovacej hlavičky), ale žiadne iné polia z vnútornej IP hlavičky (ani zdrojová a cieľová IP, ani číslo portu, ani transportný protokol), čo by mohla byť nevýhoda, hlavne vtedy keď vôbec nepoužívame DSCP značkovanie, alebo keď chceme klasifikovať aj podľa iných polí ako je DSCP.

Smerovače na konci tunela, môžu vykonať inteligentnejšie rozhodnutia o priorite paketov (oproti smerovačom vo vnútri tunela - tie vidia iba vonkajšie tunelovacie IP hlavičky - u nás je to iba R2), pretože majú prístup aj k vnútorným hlavičkám paketov predtým, ako sú enkapsulované s novou vonkajšou tunelovacou IP hlavičkou. A práve pomocou "QoS pre-classify" rozhodnem, či chcem, aby daný smerovač na konci tunela, mal možnosť prístupu k vnútornej IP hlavičke /QoS pre-classify zapnuté/, alebo nie /QoS pre-classify vypnuté/.

**Príprava**

Pri tejto topológii budeme potrebovať vygenerovať tri toky dát, preto je potrebné si pripraviť generovanie paketov prostredníctvom D-ITG generátora, spolu so zapnutím funkcie logovania v D-ITG (Logger) aj na strane príjemcu aj odosielateľa . Pakety budeme generovať z počítača PC1 na počítač PC2:

1. Tok1\_vocie: použi prednastavený z D-ITG, kodek G.711, 2 vzorky na paket
2. Tok2\_gaming: použi prednastavený z D-ITG
3. Tok3\_data: UDP, pakety veľkosti 700 B

Použi pre tieto tri toky takéto tri varianty (dopočítaj potrebnú intenzitu pre dátový tok):

1. intenzita súčtového toku (všetky 3 toky spolu): 75% z prenosovej kapacity linky
2. intenzita súčtového toku (všetky 3 toky spolu): 100% z prenosovej kapacity linky
3. intenzita súčtového toku (všetky 3 toky spolu): 125% z prenosovej kapacity linky

Dajte pozor keď budete na záver vyhodnocovať zozbierané údaje z D-ITG - potrebujete na jednej stanici mať aj logovací súbor od príjemcu aj odosielateľa.

**1. Krok: Základná konfigurácia**

Nakonfigurujte topológiu podľa hore uvedeného grafu topológie.

Ďalej je potrebné aby na sériovom R1 rozhraní bol frontovací mechanizmus nastavený na stratégiu WFQ:

R1# show interface serial0/0/0 | include Queueing



V prípade ak sa vo výpise tohto príkazu objaví FIFO, zmeňte ho na WFQ, resp. CBWFQ:

policy-map WFQ

class class-default

fair-queue

in s0/0/0

service-policy output WFQ

Ďalej si pripravte policy-map ZNACKOVANIE, v ktorej budete značkovať pakety na R1, pomocou DCSP hodnôt:

* Pakety toku 1 dostanú značku EF
* Pakety toku 2 dostanú značku AF11
* Pakety toku 3 budú ako best-effort

Túto policy-map zatiaľ neaplikujte, iba si ju pripravte.

**2. Krok: Konfigurácia statického smerovania**

Nakonfigurujte smerovačom R1, R2 predvolené cesty k smerovaču R2:

R1(config)# ip route 0.0.0.0 0.0.0.0 192.168.12.2

R3(config)# ip route 0.0.0.0 0.0.0.0 192.168.23.2

R2 bude pre sieť 172.16.0.0/16 úplne neviditeľný, bude vedieť smerovať iba do sietí 192.168.12.0/24 a 192.168.23.0/24.

**3. Krok: Konfigurácia GRE Tunelu**

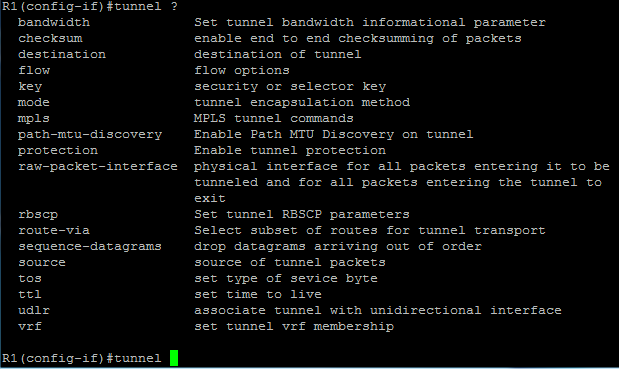
Využitie GRE Tunela si môžeme predstaviť na nasledujúcom príklade. Naša spoločnosť udržuje GRE Tunel cez smerovač internetového poskytovateľa (R2), ktorý rozdeľuje naše smerovače R1 a R3 (viď Graf Topológie). Vytvoríme tunelové rozhranie na oboch našich smerovačoch a použijeme IP adresy z adresného rozsahu 192.168.0.0 ako koncové body tunelu. Použijeme IP adresy z rozsahu 172.16.13.0/24 pre adresovanie tunelovacieho rozhrania. Smerovač R2, keďže nepatrí našej spoločnosti, nepotrebuje mať smerovacie informácie o sieťových adresách, ktoré používame v našej privátnej sieti (172.16.0.0/16).

Vytvoríme GRE Tunel zadaním príkazu **interface tunel** *číslo* pre vstup do konfiguračného módu pre tunelocacie rozhranie. Číslo tunelovacieho rozhrania má len lokálny význam, ale radšej kvôli jednoduchosti budeme používať číslo 0 pre tunelové rozhrania na oboch smerovačoch. Ďalej nakonfigurujeme adresovanie pre tunelové rozhranie príkazom **ip address** *ip\_adresa maska\_siete.* Nakoniec pridelíme zdrojovú a cieľovú adresu príkazom **source address** *adresa* a **destination address** *adresa.* Zdroj tunela môže byť podľa možnosti špecifikovaný aj rozhraní, cieľ IP adresou.

Treba si uvedomiť, že keď smerovač ide smerovať pakety, ktoré idú do vzdialenej siete za R3, v smerovacej tabuľke má uvedené, že to má poslať daným tunelovacím rozhraním. To ďalej znamená, že pridá takémuto paketu vonkajšiu tunelovaciu hlavičku, v ktorej bude source IP = začiatok tunela, destinationIP = koniec tunela, a podľa týchto IP adries sa smeruje daný paket.

R1(config)# interface tunnel 0

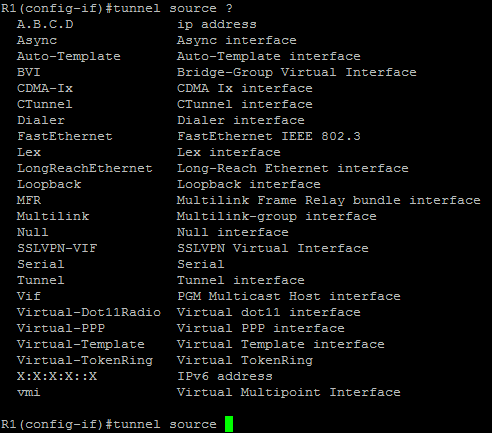
Pre zobrazenie rôznych možností príkazu tunnel použijeme operátor „?“

Poznámka: Nepotrebujeme konfigurovať tunelový mód, pretože predvolený tunelový mód je GRE.

Z ponúkaných možností vyberieme „source“:

R1(config-if)# tunnel source serial 0/0/0

Tiež pre zobrazenie rôznych možností príkazu tunnel source môžeme použiť operátor „?“



Z ponúkaných možností vyberieme serial 0/0/0.

R1(config-if)# tunnel destination 192.168.23.3

R1(config-if)# ip address 172.16.13.1 255.255.255.0

R3(config)# interface tunnel 0

R3(config-if)# tunnel source serial 0/0/1

R3(config-if)# tunnel destination 192.168.12.1

R3(config-if)# ip address 172.16.13.3 255.255.255.0

Overte konektivitu – ping z R1 na R3.

**4. Krok: Konfigurácia smerovania**

Nakonfigurujeme smerovanie (mohli by sme dynamické, ale chceme to celé pochopiť, preto použijeme statické) – treba zapezpečiť, aby pakety do LAN za R3 chodili z LAN za R1 tunelom.

**5. Testovacie scenáre**

**V každom scenári chceme sledovať**:

* 1. Spravte pre každý scenár 3 varianty: súčtový tok 75%, 100%, 125% (popis bol vyššie).
  2. Pomocou port mirroringu preskúmajte pakety, ktoré chodia z R2 na R3, pozrite si hlavičku IP paketu (IPadresy, DSCP,..) aj tunelovaciu hlavičku (IP, DSCP,..)
  3. Zaznamenajte potrebné štatistiky, vyhodnoťte výsledky
     1. WS: IO graphs
     2. D-ITG Logger: jitter pre voice prevádzku
     3. Straty pre jednotlivé 3 toky prevádzky - odkiaľkoľvek

**Scenáre:**

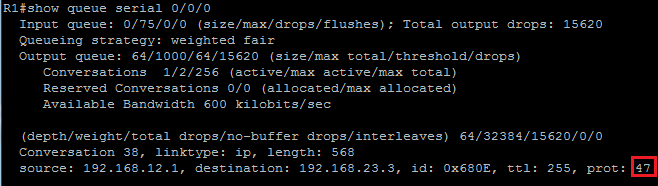
1. Scenár **bez QoS** **preclassify**
   1. Značkovanie paketov (EF, ...) ponechajte
   2. Značkovanie paketov zrušte a ponechajte toky generované bez značiek  
      Tu by sme mali vidieť, že vonkajšia tunelovacia značka tiež nemá DSCP, a rozlíšenie tokov pre QoS možno robiť len na základe iných info z hlavičky - napríklad typu protokolu, čísla portu... - tieto info sa ale do vonkajšej hlavičky neprenášajú !!! a keďže sa neprenášajú, WFQ bude mať pre best effort iba jednu frontu, do ktorej sa budú radiť pakety zo všetkých troch generovaných tokov, a výsledný jitter pre voice by mal byť pomerne veľký (aj oneskorenie keby sme ho vedeli rozumne a presne merať). Preto následne otestujeme scenár 2, kedy zapnemeQoS preclassify, ktoré nám situáciu vyrieši:
2. Scenár **s QoS** **preclassify**

Tu očakávame, že to dopadne inak ako v predošlom scenári, zdokumentujte a vyhodnoťte ako.

**Niekoľko poznámok a inštrukcií k scenáru 2:**

Zadáme príkaz show queue serial 0/0/0 pre zobrazenie aktuálne otvorených konverzácií:

R1 show queue serial 0/0/0



Všimnime si, že aktuálne sa tam nachádza len jedna konverzácia. Číslo protokolu na konci je 47, čo je predvolená tunelová enkapsulácia GRE.

QoS Pre-Classify dovoľuje, aby sieťová prevádzka bola klasifikovaná fyzickým rozhraním s frontovou stratégiou založenou na príslušnosti k toku, ešte predtým ako bude enkapsulovaná. To znamená, že šírka pásma, ktorú ponúkajú fyzické rozhrania môže byť spravodlivo rozdelená medzi odlišné tunelované toky, a nie len tie tunelované toky, ktoré budú založené na IP priorite. Toto spôsobí, že neprimerané množstvo tunelovanej sieťovej prevádzky nebude zahadzované alebo podstatne oneskorené na fyzických rozhraniach.

Povolíme QoS pre-classify vlastnosť príkazom qos pre-classify v konfiguračnom móde pre tunelové rozhranie.

R1(config)# interface tunnel 0

R1(config-if)# qos pre-classify

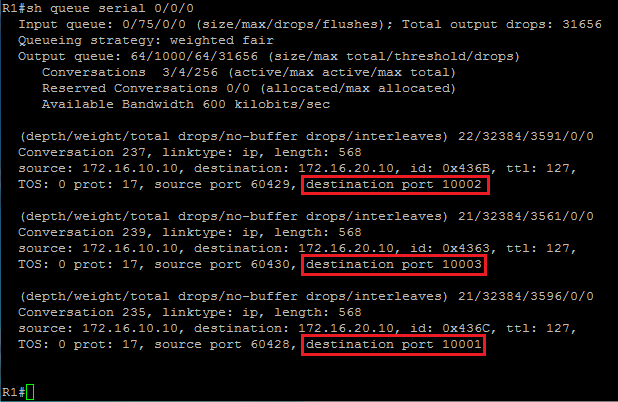
R3(config)# interface tunnel 0

R3(config-if)# qos pre-classify

Následne počas generovania sieťovej prevádzky, sa skúsime pozrieť na obsah frontu na sériovom rozhraní. Môžeme zbadať, že pre každý nami vygenerovaný prúd dát vytvorila stratégia WFQ konverzáciu. To znamená, že aj napriek tomu, že paket je zapuzdrený GRE enkapsuláciou, dôležité značky z IP paketu sú prenesené aj do tunelovacej hlavičky. Z tohto dôvodu je paketom možné poskytnúť QoS.

Následne je potrebné spustiť generovanie pripravenej sieťovej prevádzky. A potom zadáme príkaz (pozor na novších smerovačoch už nefunguje, teba pozriet inak):

R1# show queue serial0/0/0



Po prebehnutí experimentu si môžeme pozrieť štatistiky poslaných paketov:

