**Zadanie 10 – GRE tunel, QOS pre-classify**

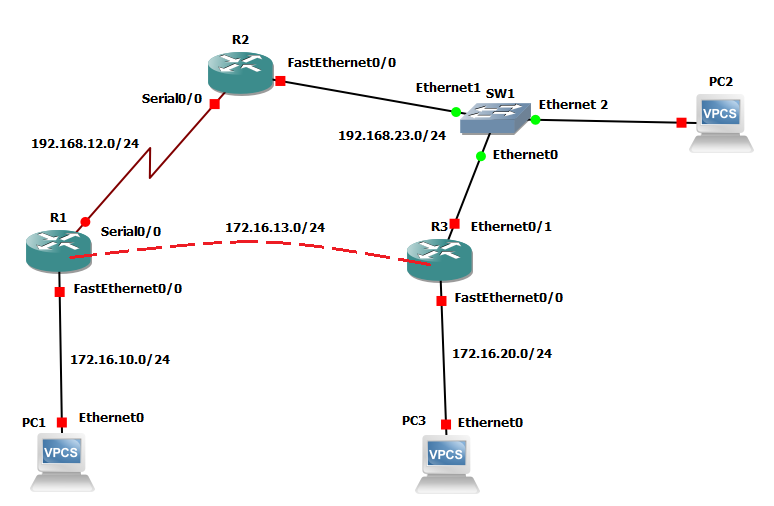
**Vladimír Jančich, Roman Kasala**

# Východisková topológia

Trojica smerovačov R1, R2 a R3 prepojených navzájom. K smerovaču R1 bol k rozhraniu fa0/0 pripojený počítač s IP adresou 172.16.10.10/24. Smerovač R1 bol sériovým káblom spojený so smerovačom R2 cez rozhranie s0/0/0 na oboch koncoch. Toto rozhranie malo limitovanú šírku pásma na 128 Kbps a tak isto aj clock rate na 128 000 b. Tieto rozhrania mali IP adresy z rozsahu 192.168.12.0/24. Smerovač R2 bol pripojený k prepínaču. Ten vytváral sieť 192.168.23.0/24 a k prepínaču bol pripojený počítač 2. Za smerovačom R3 bola sieť 172.16.20.0/24, v ktorej bol k rozhraniu fa0/0 smerovača R3 pripojený počítač 3.

Medzi smerovačmi R1 a R3 bude vytvorený GRE tunel. Na strane R1 bude mať IP adresu 172.16.13.1/24 a u R3 bude mať IP adresu 172.16.13.2/24.

PC1 bude v zadaní vystupovať v roli odosielateľa, PC3 bude príjemcom dát.



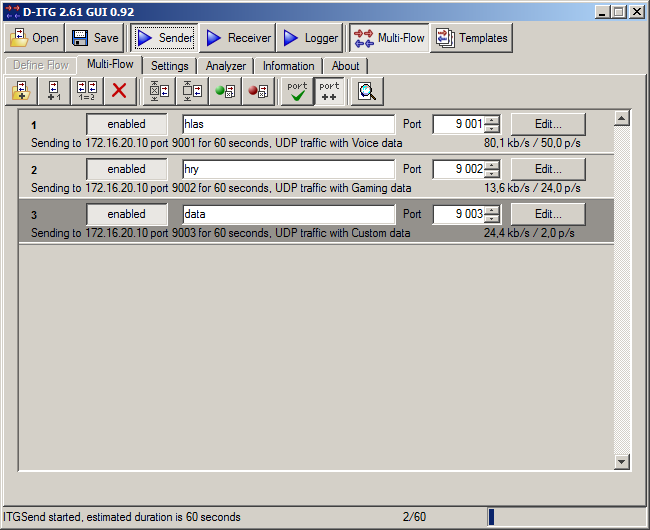
# Toky

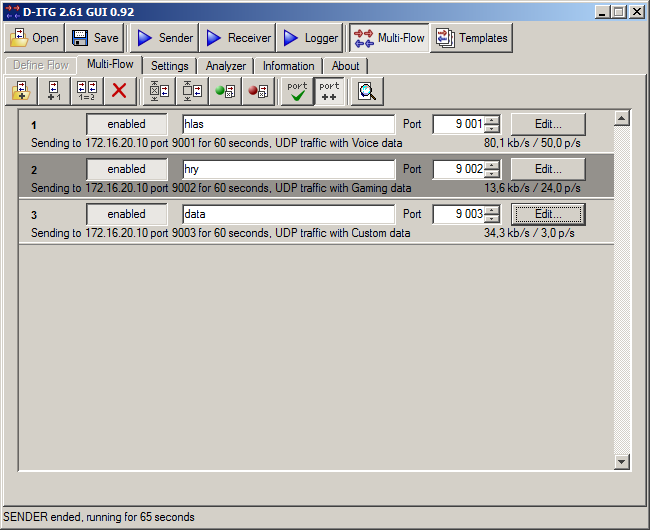
3 toky:

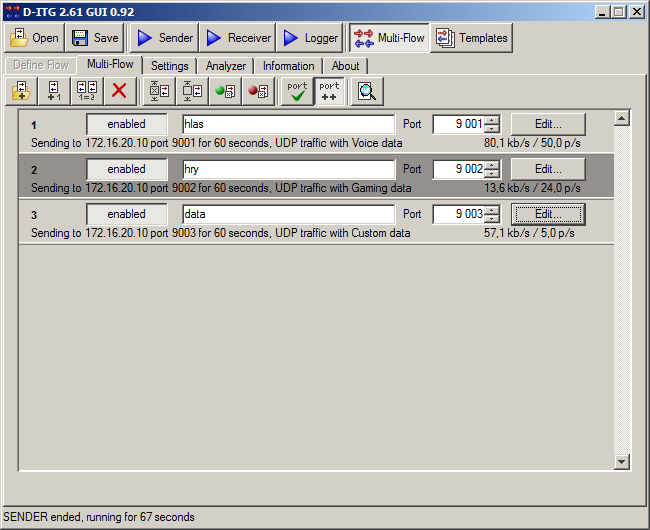
1. Prednastavený profil v D-ITG pre voice s tým, že sme použili kodek G.711 a 2 pakety na vzorku
2. prednastavený profil pre Gaming v D-ITG. Použitou hrou bol Counter Strike: Global Offensive
3. UDP tok s veľkosťou paketov 1 500 B

Vždy budeme spúšťať všetky tri toky naraz a budeme používať 3 scenáre:

1. Súčet tokov bude dosahovať asi 75 % z kapacity linky
2. Súčet tokov bude dosahovať 100 % z kapacity linky
3. Súčet tokov bude dosahovať asi 125 % z kapacity linky







# Konfigurácia WFQ a značkovania

Pre rozhranie s0/0/0 na smerovači R1 sme zmenili typ frontu z FIFO na WFQ. Stačilo vytvoriť patričnú policy-map sériou týchto príkazov:

*policy-map WFQ*

*class class-default*

*fair-queue*

Vytvorenú policy-mapu sme aplikovali na rozhranie s0/0/0 vo výstupnom smere:

*service-policy output WFQ*

V tomto kroku budeme značkovať rámce. Odlišujeme ich podľa portu pomocou access-listov:

*access-list 101 permit udp any any eq 9001*

*access-list 102 permit udp any any eq 9002*

*access-list 103 permit udp any any eq 9003*

Toky, ktoré chceme značkovať filtrujeme v class-mapách:

*class-map match-all TOK1*

*match access-group 101*

*class-map match-all TOK2*

*match access-group 102*

*class-map match-all TOK3*

*match access-group 103*

Samotné značkovanie nakoniec robíme v policy-mape. Prvý tok dostane značku EF, druhý AF11 a tretí je best-effort:

*policy-map ZNACKUJ*

*class TOK1*

*set dscp ef*

*class TOK2*

*set dscp af11*

*class TOK3*

*set dscp default*

Vytvorenú policy-mapu aplikujeme na rozhranie fa0/0 smerovača R1 vo vstupnom smere:

*service-policy input ZNACKUJ*

# Konfigurácia statického smerovania

V sieti nastavíme default route zo smerovačov R1 a R3 na R2:

*ip route 0.0.0.0 0.0.0.0 192.168.12.2 (pre R1)*

*ip route 0.0.0.0 0.0.0.0 192.168.23.1 (pre R3)*

Následne ešte nastavíme statické smerovanie do LAN zo smerovača R1 a R3. Smerovanie sa bude diať cez GRE tunel, nie cez smerovač R2.

*ip route 172.16.20.0 255.255.255.0 172.16.13.2 (pre R1)*

*ip route 172.16.10.0 255.255.255.0 172.16.13.1 (pre R3)*

# Konfigurácia GRE tunelu

Medzi smerovačom R1 a R3 sme vytvorili GRE tunel:

PRE R1:

*interface Tunnel0*

*ip address 172.16.13.1 255.255.255.0*

*tunnel source 192.168.12.1*

*tunnel destination 192.168.23.2*

PRE R3:

*interface Tunnel0*

*ip address 172.16.13.2 255.255.255.0*

*tunnel source 192.168.23.2*

*tunnel destination 192.168.12.1*

Po tomto kroku je už konektivita z LAN1 do LAN2 funkčná a prebieha práve cez tento tunel.

# Konfigurácia port-mirroringu

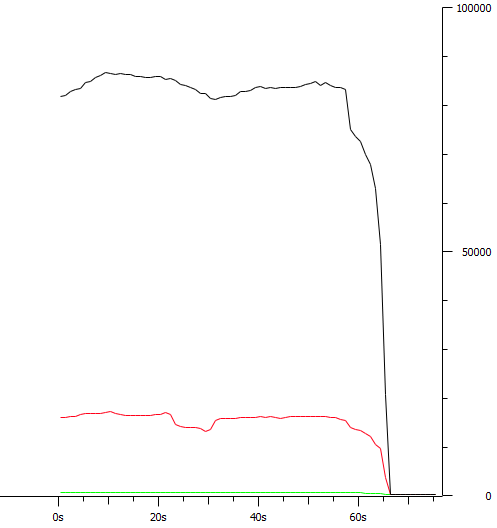
Prepínač sme nakonfigurovali pre port-mirroring, aby sme na PC3 mohli sledovať údaje v GRE tuneli a DSCP značky:

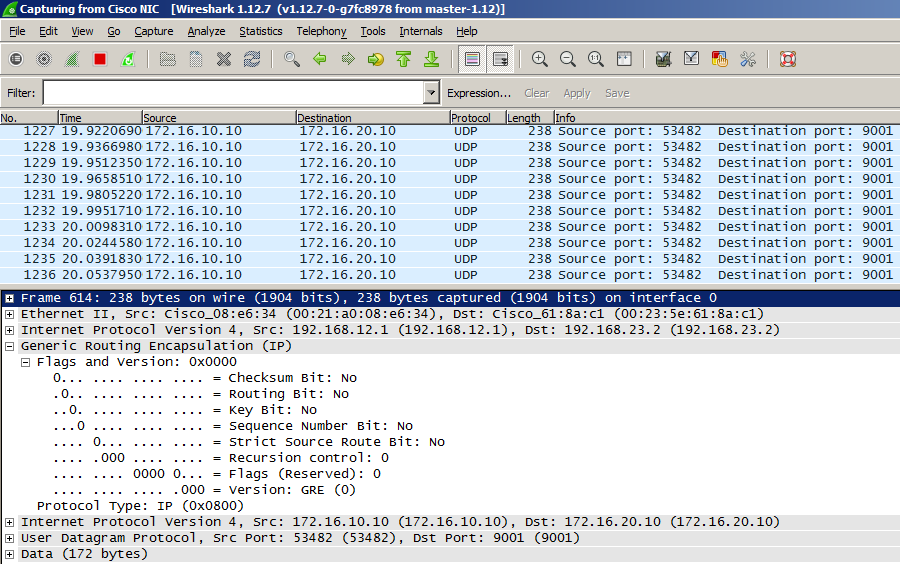
*monitor session 1 source interface Fa0/1*

*monitor session 1 destination interface Fa0/3*

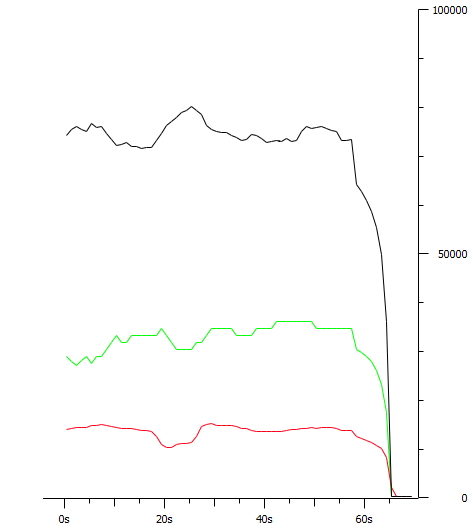
# Testy

## Scenár jeden: značkovanie, 75 % tok

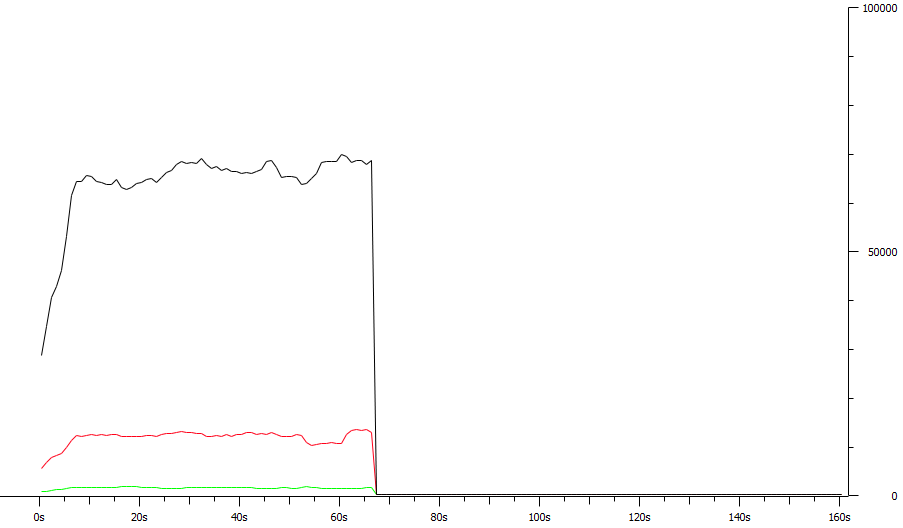




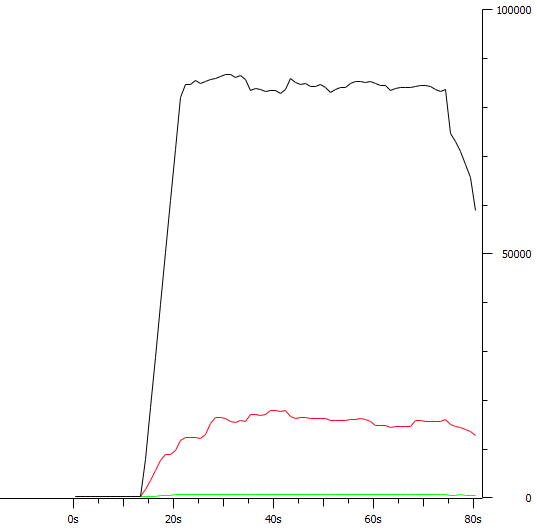
## Scenár1 – značkovanie, 100 % tok



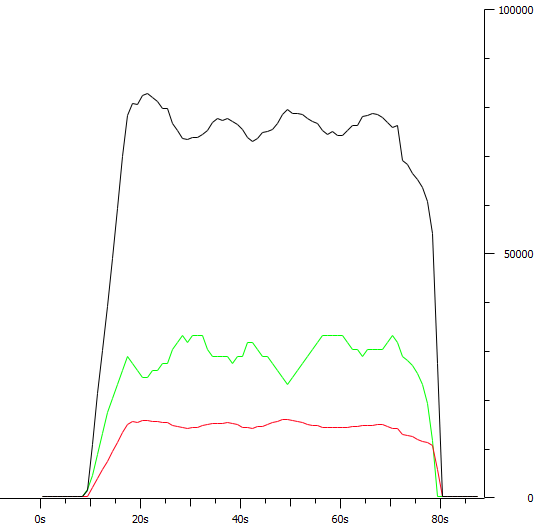
## Scenár 1 – značkovanie, 125 % tok



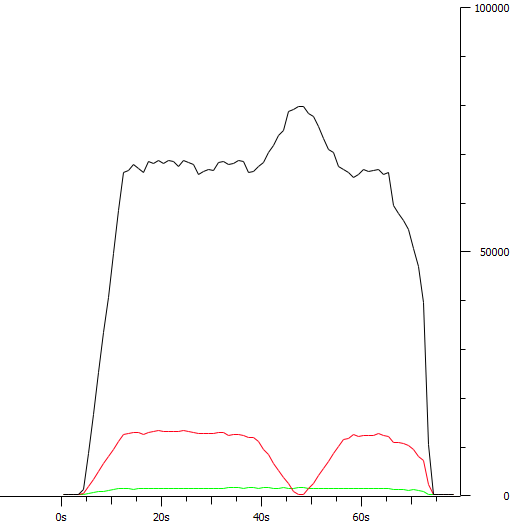
## Scenár 2 – bez značky, 75 % tok



## Scenár 2 – bez značky, 100 % tok



## Scenár 2 – bez značky, 125 % tok

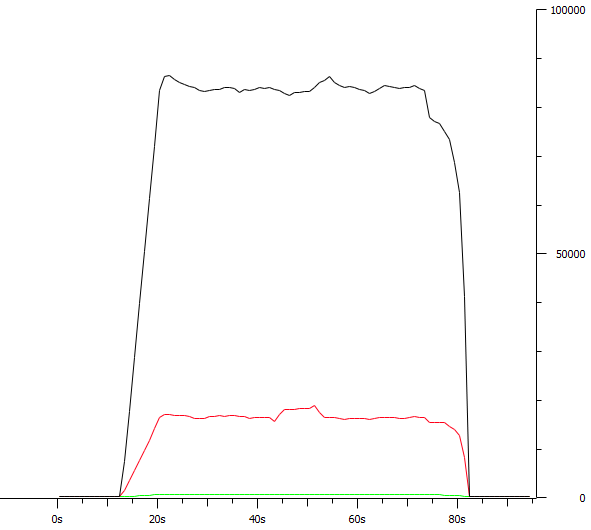


Po týchto testoch sme na smerovačoch R1 a R3 nastavili pre rozhranie tunel0 qos pre-classify a znova sme 3x spustili toky:

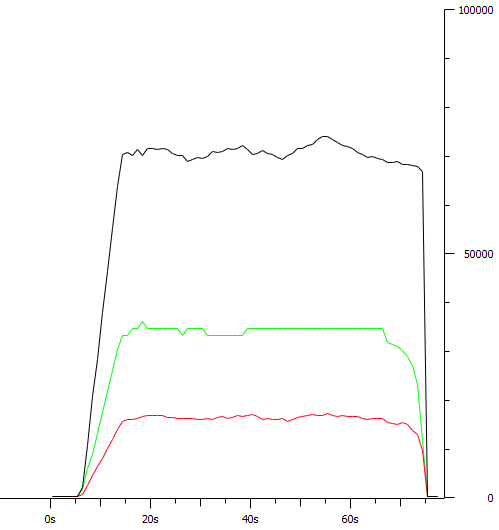
*interface Tunnel0*

*qos pre-classify*

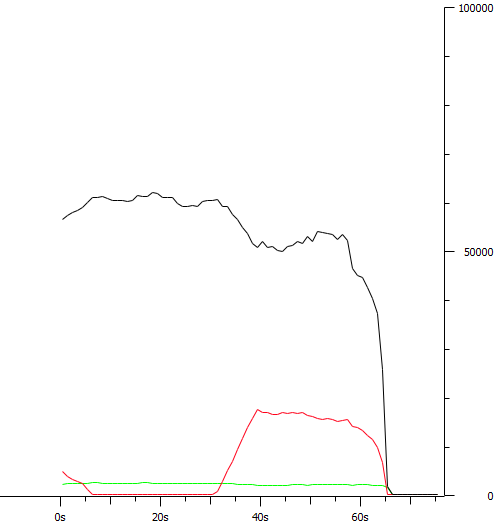
## Scenár 3 – pre-classify, bez značky, 75 % tok



## Scenár 3 – pre-classify, bez značky, 100 % tok



## Scenár 3 – pre-classify, bez značky, 125 % tok



# Dodatok

Pri sledovaní tokov cez grafy vo Wiresharku sme si všimli, že ak mal posledný tok nastavenú veľkosť 1 500 B, graf vo Wiresharku nevykresľoval dobre skutočný prenesený objem dát, ktorý sme si overili cez logger v D-ITG. Toto správanie môžeme vidieť vždy pri situácii kedy púšťame 100 % tok, kde mal posledný tok nastavenú veľkosť 1 400 B a grafy vyzerali vierohodnejšie. Príčina takéhoto správania Wiresharku je neznáma, neodporúčame však púšťať UDP tok s veľkosťou paketov 1 500 B (na počte paketov za sekundu nezáleží).