

QoS implementacija



Quality of Service predstavlja mogućnost dodjeljivanja različitih prioriteta različitim aplikacijama, korisnicima i tokovima podataka ili osiguranja određenog nivoa usluge za neki tok podataka. Implementacija **QoS** se konfigurira u **MQC** (eng. *Modular QoS Command-Line Interface*) kojom se razdvaja klasifikacija prometa od definiranja politike prometa.

QoS implementacija

Implementacija **QoS** se konfigurira u **MQC** (eng. *Modular QoS Command-Line Interface*) kojom se razdvaja klasifikacija prometa od definiranja politike prometa.

Implementacija **QoS** sa **MQC** izvodi se u tri koraka:

1. Definiranje prometa u klase ili grupe koristeći **class-map** naredbe. Cisco preporuča definiranje od 4 do 11 klasa. Konfigurira se na slijedeći način:

```
Router(config)# class-map [match-all | match-any] class-name
```

Kreira klasu nad kojom se dalje uspoređuju paketi unutar *class* mape po određenim *match* kriterijima, *match-all* (logičko i) sve usporedbe moraju biti istinite postavljeno „po defaultu” (ukoliko se ne definira), dok *match-any* (logičko ili) jedna usporedba mora biti istinita.

Dolje navedene naredbe su opcionalne i moguće je definirati više usporednih (*match*) kriterija:

```
Router(config-cmap)# match access-group {access-group | name access-group-name}
```

```
Router(config-cmap)# match any
```

```
Router(config-cmap)# match cos cos-number
```

```
Router(config-cmap)# match destination-address mac address
```

```
Router(config-cmap)# match [ip] dscp dscp-value
```

```
Router(config-cmap)# match mpls experimental mpls-values
```

```
Router(config-cmap)# match [ip] precedence precedence-value
```

```
Router(config-cmap)# match source-address mac address-destination
```

```
Route(config-cmap)# exit
```

2. Definiranje QoS politike za definirane prometne klase koristeći policy-map naredbu. Ovaj korak spaja QoS značajke sa prometnim klasama, policy mapa definira tretman definiranih klasa prometa. Konfigurira se na slijedeći način:

```
Router(config)# policy-map policy-name
```

Kreira ime za policy mapu i ulazi u policy-map configuration mode

```
Router(config-pmap)# class {class-name | class-default}
```

Specificira ime maloprije kreirane prometne klase i ulazi u *policy-map class configuration mode*

Dolje navedene naredbe su opcionalne i moguće je definirati više značajka **QoS**:

```
Router(config-pmap-c)# bandwidth {bandwidth-kbps | percent percent}
```

Specificira najmanju garantiranu propusnost prometnoj klasi u periodu zagušenja, moguće je definirati u *kbps* ili u postocima ukupne dostupne propusnosti.

```
Router(config-pmap-c)# fair-queue number-of-queues
```

Specificira broj rezerviranih redova čekanja (*queue*) za prometnu klasu.

```
Router (config-pmap-c)# police cir bps [bc burst-normal] [be burst-max] conform-  
action action  
exceed-action action [violate-action action]
```

Konfiguriranje reguliranja prometa (*traffic policing*)

```
Router(config-pmap-c)# priority {bandwidth-kbps | percent percentage} [burst]
```

Postavlja prioritet prometnoj klasi koja pripada *policy* mapi

```
Router(config-pmap-c)# queue-limit number-of-packets
```

Specificira ili modificira maksimalni broj paketa koji može sadržavati red čekanja za prometnu klasu unutar *policy* mape

```
Router(config-pmap-c)# set cos {cos-value | from-field [table table-map-name]}
```

Postavlja vrijednost *Layer 2 Class of Service (CoS)* na izlazni paket.

```
Router(config-pmap-c)# set [ip] dscp {dscp-value | from-field [table table-map-name]}
```

Markiranje paketa postavljanjem **DSCP** (eng. *differentiated services code point*) vrijednosti unutar **ToS** (eng. *Type of Service*) bajta.

```
Router(config-pmap-c)# set precedence {precedence-value | from-field [table table-map-name]}
```

Postavlja *precedence* vrijednost unutar zaglavlja paketa

```
Router(config-pmap-c)# set mpls experimental value
```

Postavlja odnosno markira pakete *mpls* eksperimentalnim bitovima koji su usporedni sa *policy* mapom

```
Router(config-pmap-c)# shape {average | peak} cir [bc] [be]
```

Oblikuje promet po specificiranim normama i algoritmu

```
Router(config-pmap-c)# service-policy policy-map
```

Veže *service-policy* sa *Class Based Weighted Fair Queue (CBWFQ)* na klasu.

3. Primjena definiranih politika u ulaznom ili izlaznom smjeru sučelja ili pod sučelja koristeći *service-policy* naredbu.

```
Router(config)# interface interface
```

```
Router(config-if)# service-policy output {input | output} policy-map-name
```

Naredbe za provjeru prometnih klasa i politika

```
Router# show class-map [type {stack | access-control}] [class-map-name]
```

Prikazuje sve *class* mape i njihove usporedne kriterije

```
Router# show policy-map policy-map class class-name
```

Prikazuje konfiguraciju određene klase unutar *policy* mape

```
Router# show policy-map policy-map
```

Prikazuje konfiguraciju svih klasa za određenu *policy* mapu ili sve klase za sve *policy* mape

```
Router# show policy-map interface [type access-control] type number [vc [vpi/] vci] [dlci dlci] [input | output]
```

Prikazuje statistike paketa svih klasa koje su konfigurirane za sve *service* politike na određenom sučelju ili podsučelju ili na **PVC** (*permanent virtual circuit*) sučelju.

pon, 2009-03-09 21:55 - Miroslav Božić **Kategorije:** [Mreža](#) [1]

Vote: 0

No votes yet

Source URL: <https://sysportal.carnet.hr/node/528>

Links

[1] <https://sysportal.carnet.hr/taxonomy/term/29>