

QoS in RouterOS v6.x

by Alfredo Giordano

9:00 AM

MUM ITALY 2014

About Me

- Graduate in electronic engineering
- Mikrtotik certified and consultant and trainer
- Working telecommunication since 2001
- RIPE member
- ISP CEO and designer
- Specialized in Routing, QoS, WAN access, wireless

For an ISP what is QoS about?

- QoS is about Bandwidth management
- QoS is about enabling certain type of services
- QoS is about guarantee certain level of services
- QoS is about well manage what we have
- QoS is about keeping customers happy!

Identifying the problem(s)

Identifying the problem(s)

- Basically we can split the problem in two parts:
 - Limit the available bandwidth per user (or per type of contract subscribed)
 - Make sure that certain types of services will be provided with priority respect to others

Knowing the tools:

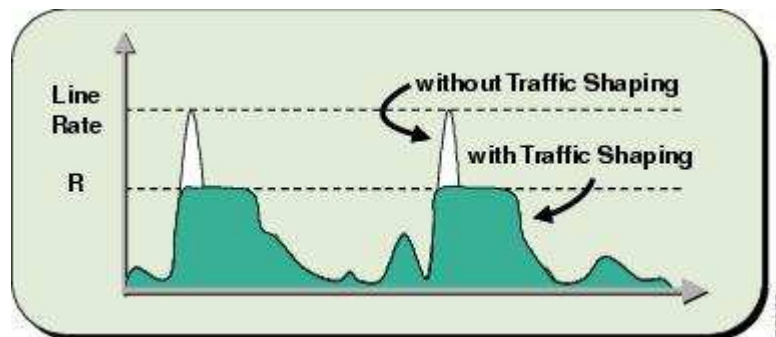
Fundamentals

- Traffic control is done on the outbound interface (we have no control on how much traffic is being sent to us)
- Rate limit is done by dropping some low priority packets so we have capacity for higher priority packets
- We need to know how much bandwidth is available
- WE ARE NOT REORDERING PACKETS, packets will leave the router in the exact sequence as they are received (provided that we are forwarding them)

Fundamentals

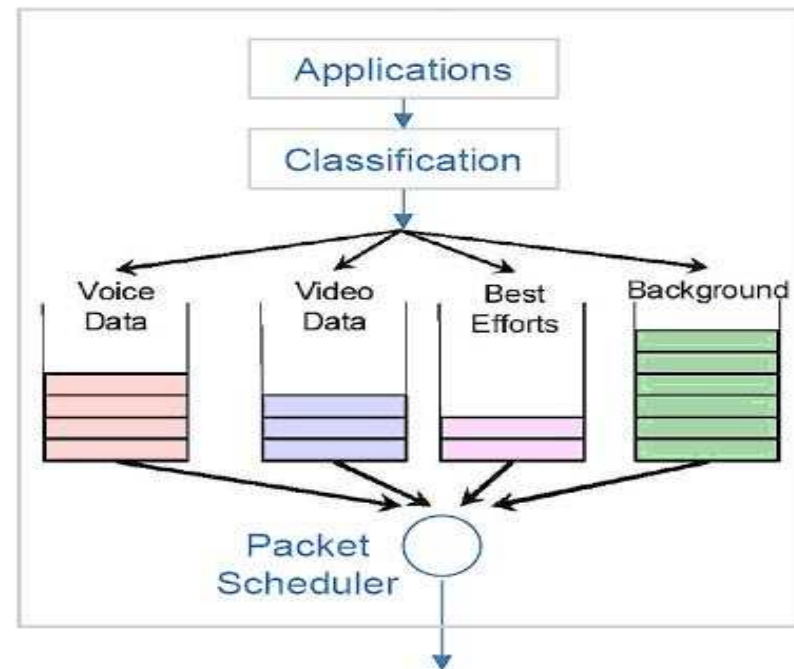
Traffic Shaping

- Limits the transmit rate of traffic to a certain value by temporary buffering exceeding packets:



Traffic Priority

- Classify traffic based on application.



Identifying the solution

Identifying the solution

- We need **two** QoS facilities:
 - First to classify the traffic:
 - Let flow the high priority packets
 - Drop low priority packets if they are coming too fast
 - Second to shape the traffic:
 - Avoid one user to monopolize the available bandwidth
 - Sell different services with different bandwidth rates

Identifying the solution

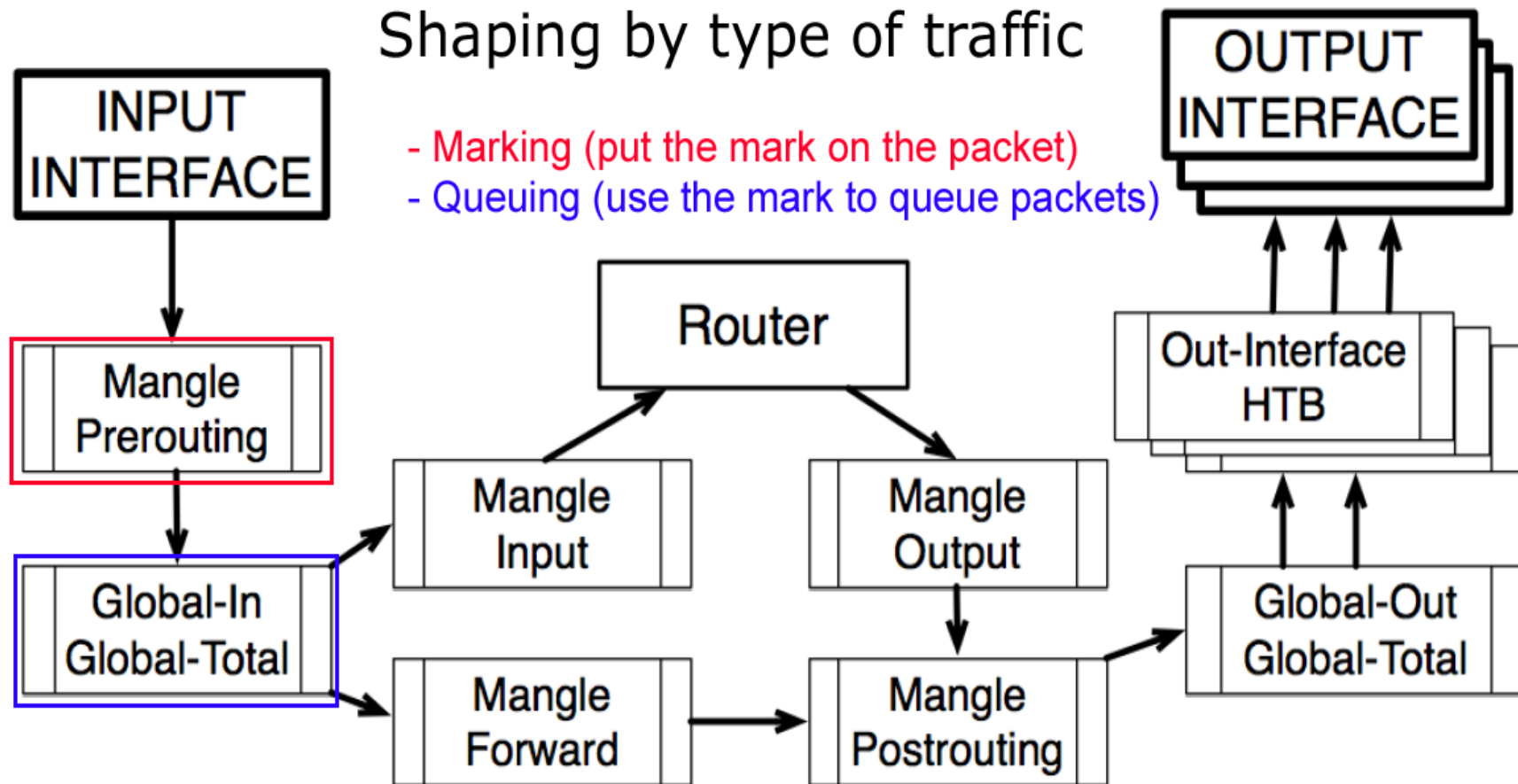
we can either:

- Use two RouterOs boxes
 - Less Hardware requirements
 - Very simple setup
- Setup double QoS on a single RouterOs box
 - Power budget (solar panel etc..)
 - Thermal budget
 - Money budget
 - Avoid a point of failure
 - Just because we can!

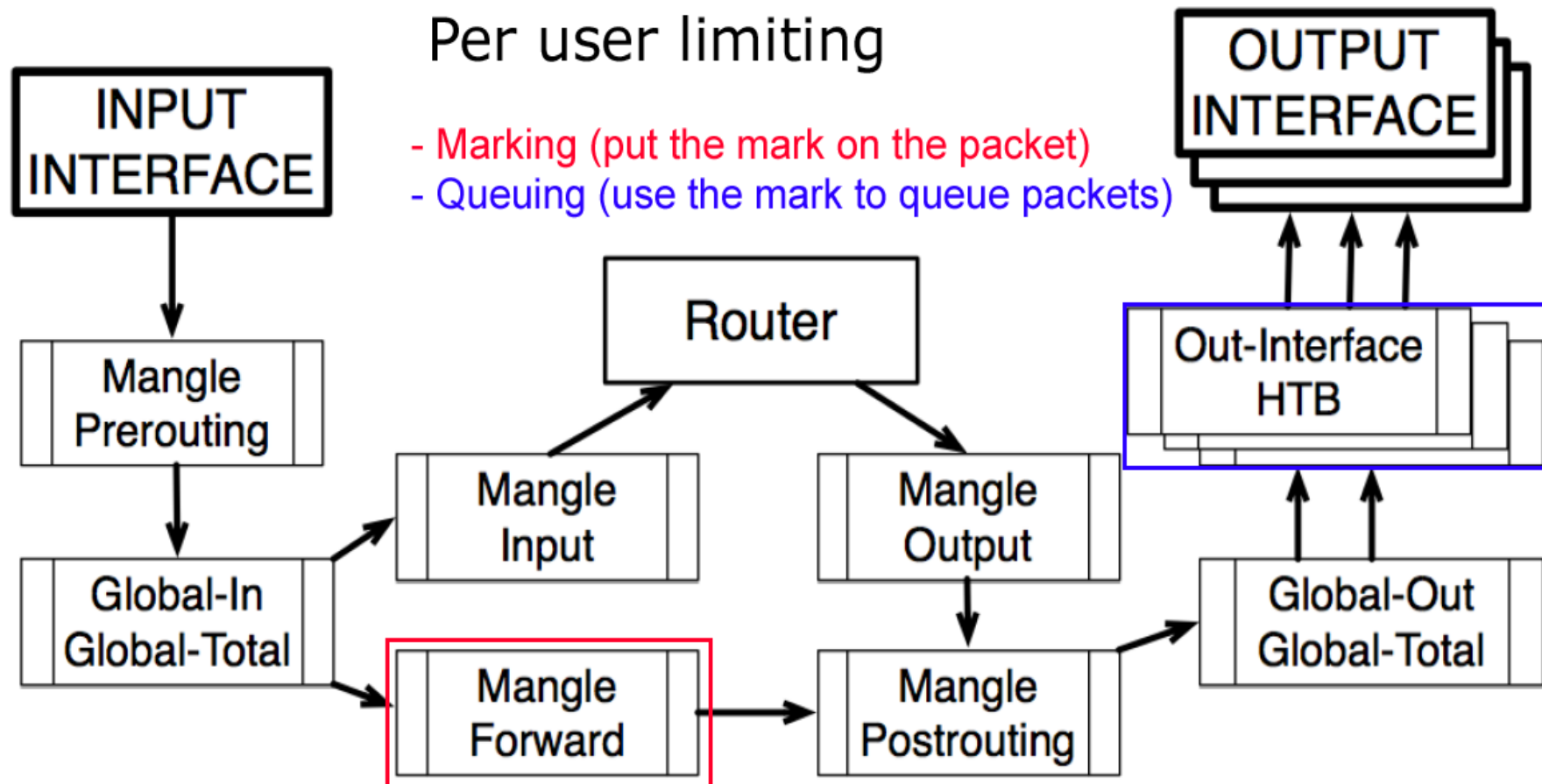
How to do it

Disclaimer: This is one way of doing it
(there are more)

Double QoS(RouterOs V5.x)



Double QoS(RouterOs V5.x)



Double QoS(RouterOs V5.x)

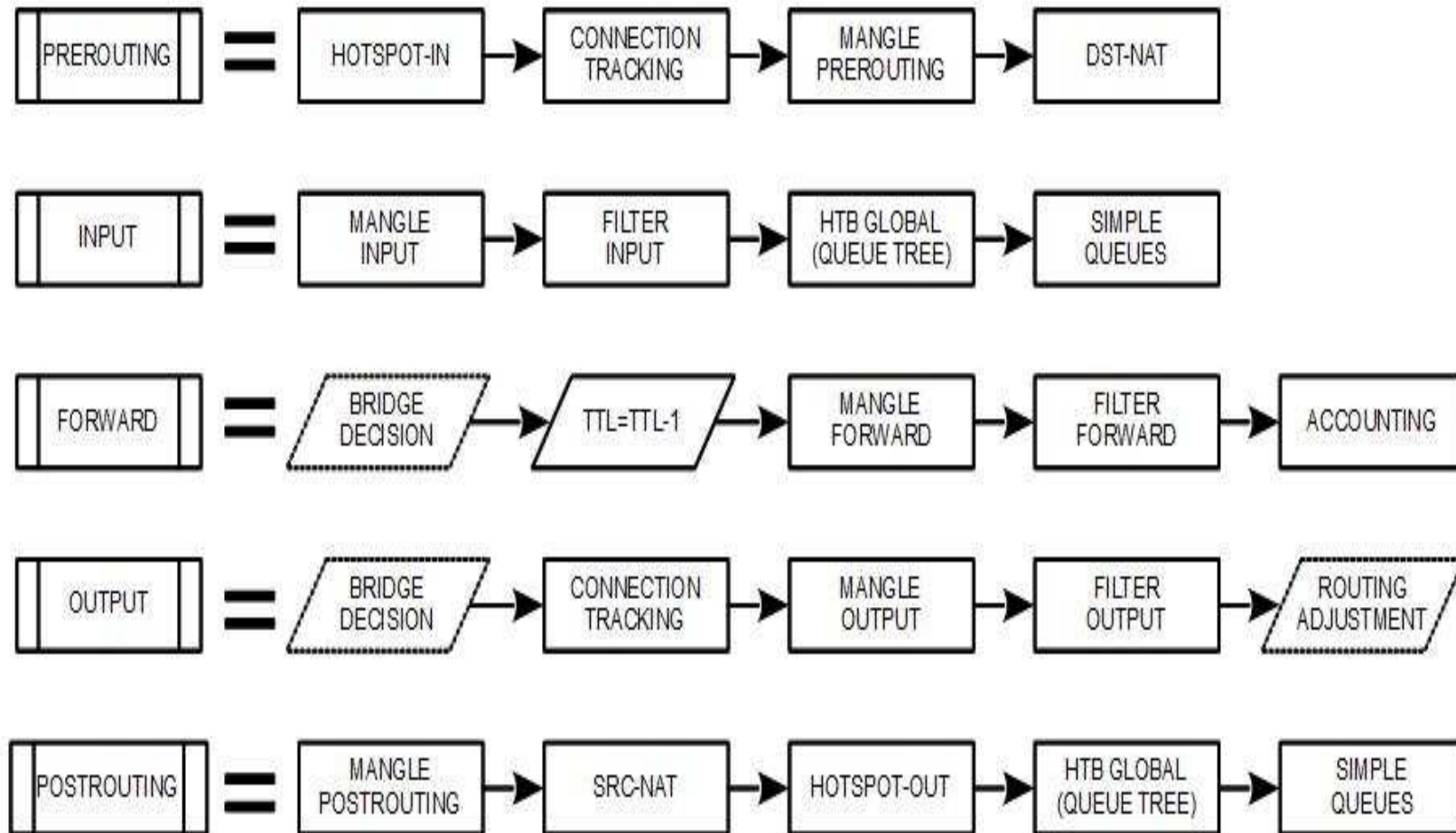
- Basically we will use the mark facility two times -

- Mark traffic by traffic type in mangle chain Prerouting
- Limit traffic by type in Global-in HTB
- Re-Mark traffic by clients in mangle chain Forward
- Limit traffic per client in Interface HTB

changes in RouterOs v6.x

- No more global-in and global-out, replaced by a "global" located just before simple queues;
- Better simple queues selection algorithm (hashing);
- Simple queues happen in different place at the very end of postrouting and input chains;
- Simple queues have separate priority setting for download/upload/total;
- Simple queues target-addresses and interface parameters are joined into one target parameter, now supports multiple interfaces match for one queue;
- Simple queues dst-address parameter is changed to dst and now supports destination interface matching;

RouterOs v6.x Packetflow



Double QoS(routerOs V6.x)

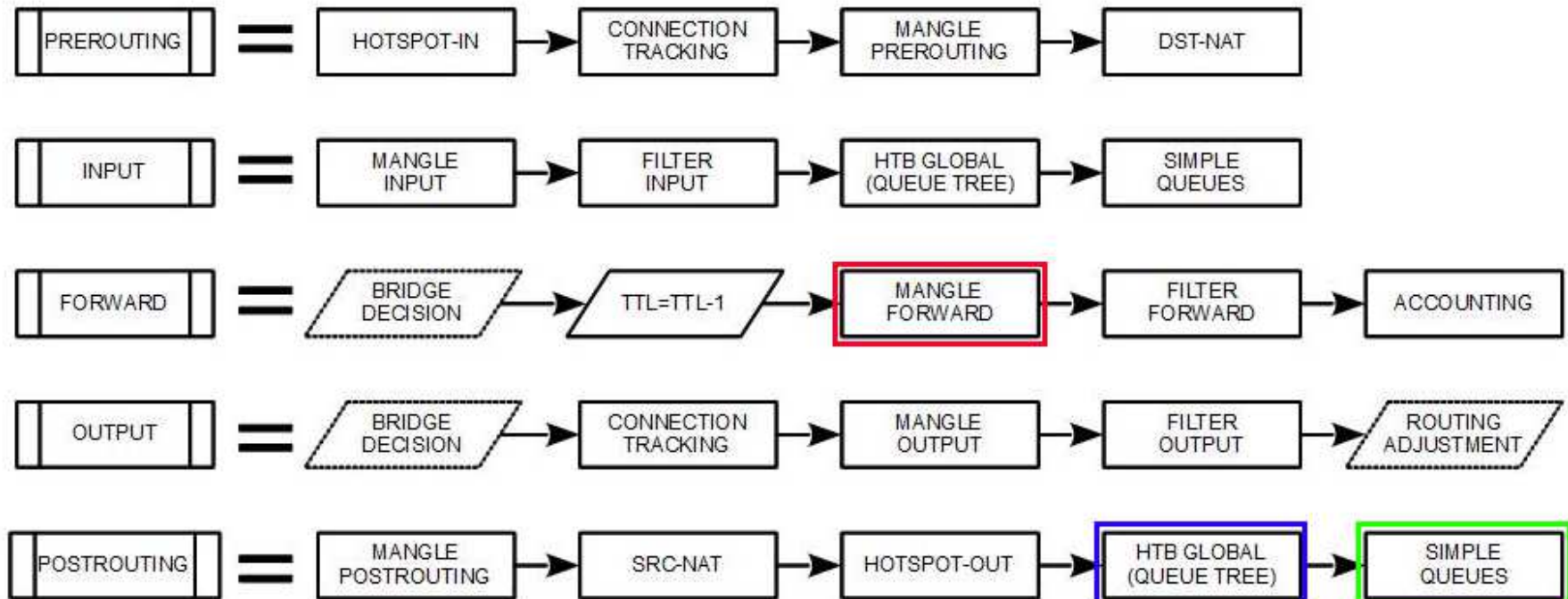
- Because queuing happens at one place we cannot mark, queue, remark and re-queue as in v5.x
- We can use mangle to mark packet by type of service and queue them in queue tree but...
- We need a separate facility to queue packets to achieve per user limitation.
- Ideas?

Simple queues

- Not for only for simple tasks anymore...
- No need to mark can identify traffic based on dst-address, interface, etc...
- Fast... especially on multicore hardware
- Number of simple queue is not relevant anymore
- We can have thousands of them and we can easily create them either dynamically or by scripts.

The big picture

- Service type marking
- Service type queuing
- User limiting queuing

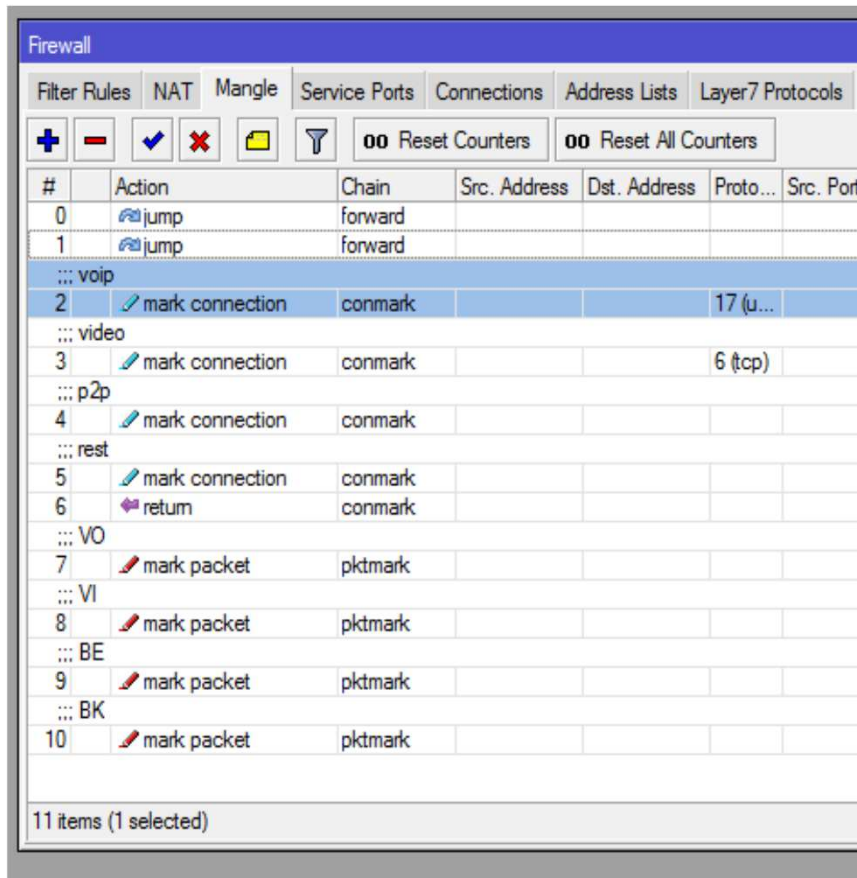


Practical example - simple PPPoE AC

- restrict bandwidth per user
- support voice
- video streaming
- ptp programs

Service type - Packet Marking

- Winbox view



#	Action	Chain	Src. Address	Dst. Address	Proto...	Src. Port
0	jump	forward				
1	jump	forward				
... voip						
2	mark connection	conmark				17 (u...
... video						
3	mark connection	conmark				6 (tcp)
... p2p						
4	mark connection	conmark				
... rest						
5	mark connection	conmark				
6	return	conmark				
... VO						
7	mark packet	pktmark				
... VI						
8	mark packet	pktmark				
... BE						
9	mark packet	pktmark				
... BK						
10	mark packet	pktmark				

11 items (1 selected)

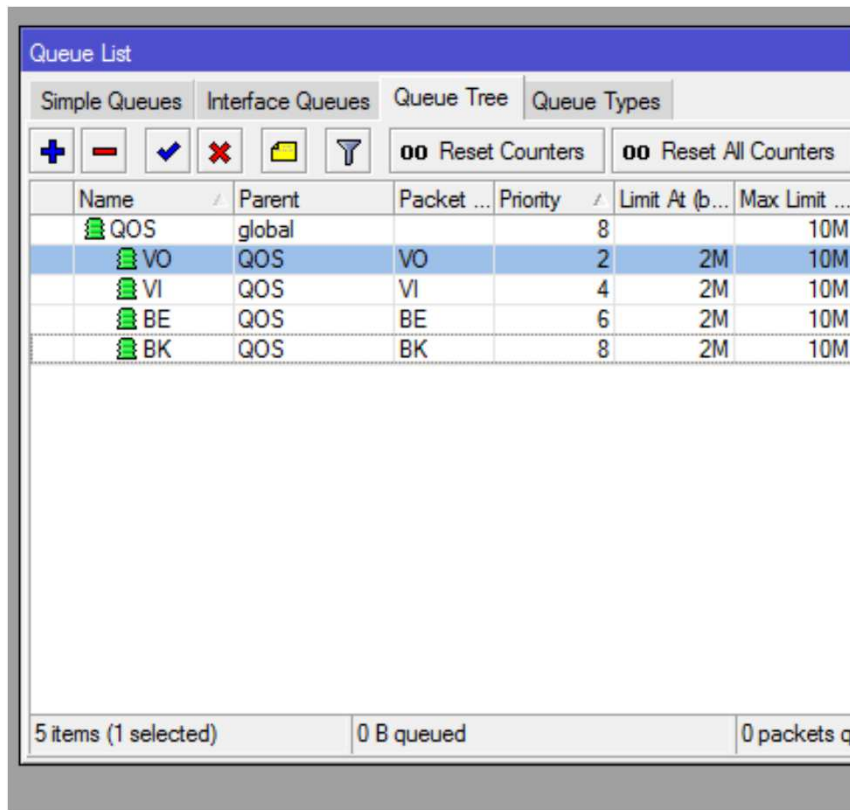
- Jump to a chain where we will put the mark on the connection
- Jump to a chain where we will put the mark on the packet (based on the connection)
- This will reduce overhead for complex matches.

Mangle - Export view

```
/ip firewall mangle
add action=jump chain=forward connection-mark=no-mark jump-target=conmark
add action=jump chain=forward connection-mark=!no-mark jump-target=pktmark
add action=mark-connection chain=conmark comment=voip dst-port=5060-
5061,16000-17000 new-connection-mark=voip-con protocol=udp
add action=mark-connection chain=conmark comment=video dst-address-
list=youtube new-connection-mark=video-con protocol=tcp
add action=mark-connection chain=conmark comment=p2p new-connection-
mark=p2p-con p2p=all-p2p
add action=mark-connection chain=conmark comment=rest connection-mark=no-
mark new-connection-mark=all-con
add action=return chain=conmark
add action=mark-packet chain=pktmark comment=VO connection-mark=voip-con
new-packet-mark=VO
add action=mark-packet chain=pktmark comment=VI connection-mark=video-con
new-packet-mark=VI
add action=mark-packet chain=pktmark comment=BE connection-mark=all-con
new-packet-mark=BE
add action=mark-packet chain=pktmark comment=BK connection-mark=p2p-con
new-packet-mark=BK
```

Service type – Queue Tree

- Winbox view



The screenshot shows the 'Queue List' window in Winbox. It has tabs for 'Simple Queues', 'Interface Queues', 'Queue Tree', and 'Queue Types'. The 'Queue Tree' tab is selected. Below the tabs are several icons and two buttons: 'Reset Counters' and 'Reset All Counters'. A table displays the queue configuration:

Name	Parent	Packet ...	Priority	Limit At (b...	Max Limit ...
QOS	global		8		10M
VO	QOS	VO	2	2M	10M
VI	QOS	VI	4	2M	10M
BE	QOS	BE	6	2M	10M
BK	QOS	BK	8	2M	10M

At the bottom of the window, it shows '5 items (1 selected)', '0 B queued', and '0 packets q'.

- Parent queue in global for upload and download traffic sets max-limit
- Child classes with higher priority will be able to reach max-limit before class with lower priority.
- Traffic queued based on the packet mark

Queue Tree - Export view

```
/queue tree
add max-limit=10M name=QOS parent=global queue=default
add limit-at=2M max-limit=10M name=BK packet-mark=BK parent=QOS
add limit-at=2M max-limit=10M name=BE packet-mark=BE parent=QOS
priority=6
add limit-at=2M max-limit=10M name=VI packet-mark=VI parent=QOS
priority=4
add limit-at=2M max-limit=10M name=VO packet-mark=VO parent=QOS
priority=2
```

Per user limit – Dynamic Simple Queue

- Winbox view

The screenshot displays the Winbox configuration interface for PPP. At the top, there are tabs for 'Interface', 'PPPoE Servers', 'Secrets', 'Profiles', and 'Active Connections'. Below these are several icons for adding, deleting, and filtering. A table lists the configured services:

Service ...	Interface	Max MTU	Max MRU	MRRU	Default Profile	Authentication
	vlan200	1492	1492		default	pap chap mschap1 mschap2

Below the table, two configuration windows are open:

- PPPoE Service <>**: Shows fields for Service Name, Interface (vlan200), Max MTU (1492), Max MRU (1492), MRRU, Keepalive Timeout (35), Default Profile (default), and authentication options (pap, mschap1, chap, mschap2).
- PPP Profile <default>**: Shows the 'Queue' tab with a Rate Limit (rx/tx) of 256k/2560k. A red arrow points from the text 'Simple queue Upload' to the '256k' part of the rate limit, and a blue arrow points from 'Simple Queue Download' to the '2560k' part.

Dynamic Simple Queue – Export view

```
/ppp profile  
set 0 dns-server=8.8.8.8,8.8.4.4 local-address=10.0.0.6 only-  
one=yes rate-limit="256k/2560k" remote-address=customers use-vj-  
compression=no
```

```
/interface pppoe-server server  
add disabled=no interface=vlan200 keepalive-timeout=35 max-  
mru=1492 max-mtu=1492 one-session-per-host=yes
```

Per user limit – Simple Queue

Queue List

Simple Queues | Interface Queues | Queue Tree | Queue Types

#	Name	Target	Upload Max Limit	Download Max ...
32	D <pppoe-w168186>	<pppoe-w168186>	256k	2560k
17	D <pppoe-w169187>	<pppoe-w169187>	256k	2560k
123	D <pppoe-w192166>	<pppoe-w192166>	256k	2560k
6	D <pppoe-w197221>	<pppoe-w197221>	256k	2560k
160	D <pppoe-w212150>	<pppoe-w212150>	256k	2560k
19	D <pppoe-w222140>	<pppoe-w222140>	256k	2560k
73	D <pppoe-w236112>	<pppoe-w236112>	256k	2560k
39	D <pppoe-w244113>	<pppoe-w244113>	256k	2560k
40	D <pppoe-w265084>	<pppoe-w265084>	256k	2560k
29	D <pppoe-w273232>	<pppoe-w273232>	256k	2560k
128	D <pppoe-w277073>	<pppoe-w277073>	256k	2560k
22	D <pppoe-w290077>	<pppoe-w290077>	256k	2560k
135	D <pppoe-w295064>	<pppoe-w295064>	256k	2560k
93	D <pppoe-w306055>	<pppoe-w306055>	256k	2560k
46	D <pppoe-w307056>	<pppoe-w307056>	256k	2560k
33	D <pppoe-w309053>	<pppoe-w309053>	256k	2560k
125	D <pppoe-w312046>	<pppoe-w312046>	256k	2560k
26	D <pppoe-w335152>	<pppoe-w335152>	256k	2560k
9	D <pppoe-w359018>	<pppoe-w359018>	256k	2560k
142	D <pppoe-w360012>	<pppoe-w360012>	256k	2560k
42	D <pppoe-w418227>	<pppoe-w418227>	256k	2560k
5	D <pppoe-w424223>	<pppoe-w424223>	256k	2560k
65	D <pppoe-w434013>	<pppoe-w434013>	256k	2560k

166 items (1 selected) 0 B queued 0 packets queued

Simple Queue <<pppoe-w424223>>

Advanced | Statistics | Traffic | Total | Total Statistics | ...

Target Upload Target Download

Rate: 35.5 kbps 2.3 Mbps

Packet Rate: 85 p/s 200 p/s

dynamic enabled

OK

Copy

Remove

Reset Counters

Reset All Counters

Torch

Sources

- QoS Best Practice and RouterOS v6 presentations by Janis Megis
- Mikrotik wiki
- QoS theory

THANK YOU

- Q&A Comments and suggestions
 - Alfredo Giordano (agiordano@warian.net)

