

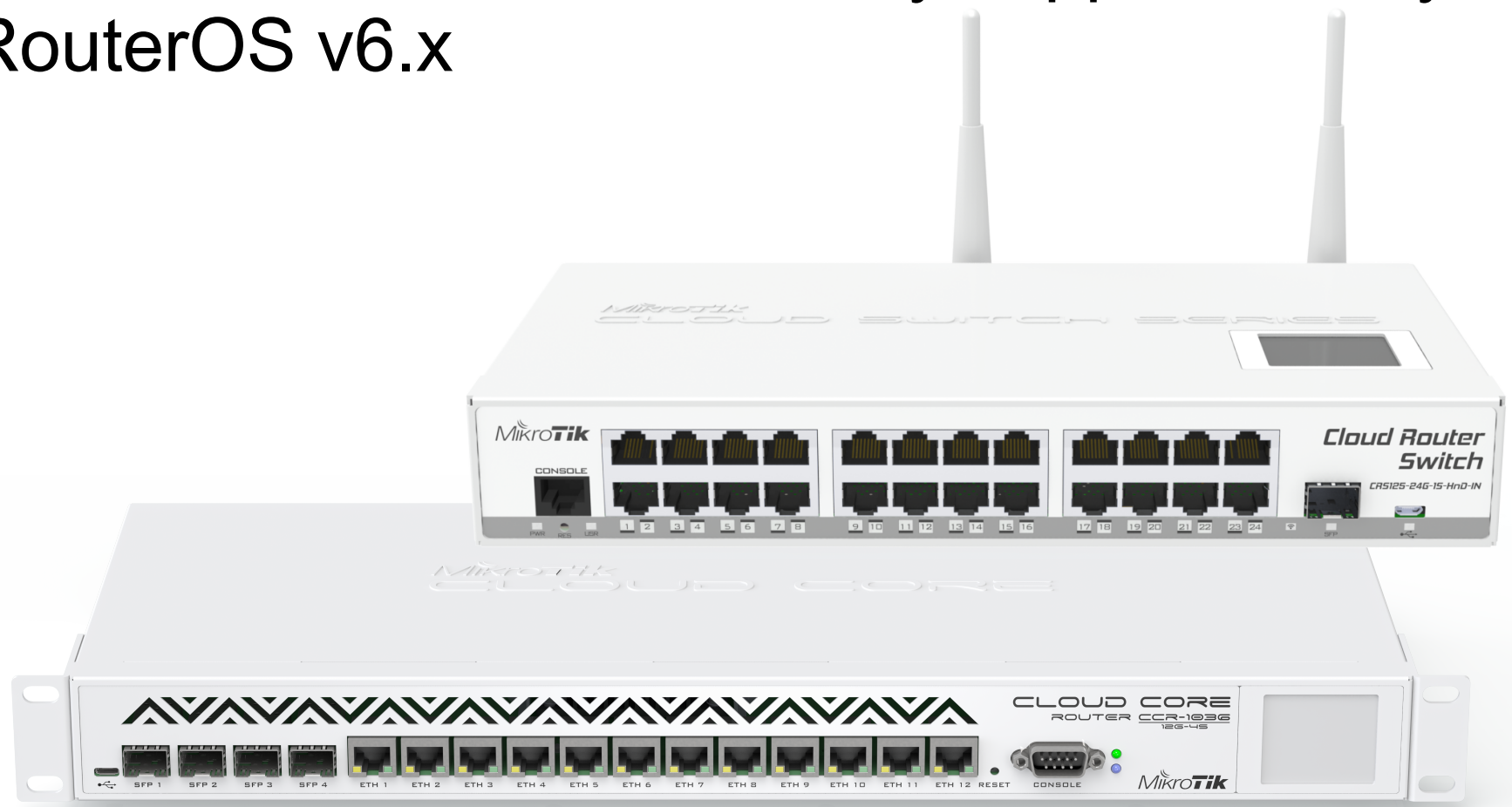
# **MikroTik RouterOS v6**

## **Whats new??**

St. Louis, MO  
MUM USA 2013

# Support for New Products

- CloudCoreRouter, CloudRouterSwitch and other new devices will be fully supported only in RouterOS v6.x



# New Linux Kernel

- RouterOS 5.25
  - Linux Kernel version 2.6.35
- RouterOS 6.x
  - Linux Kernel version 3.3.5+
- For more detailed information see:  
<http://www.kernel.org/>

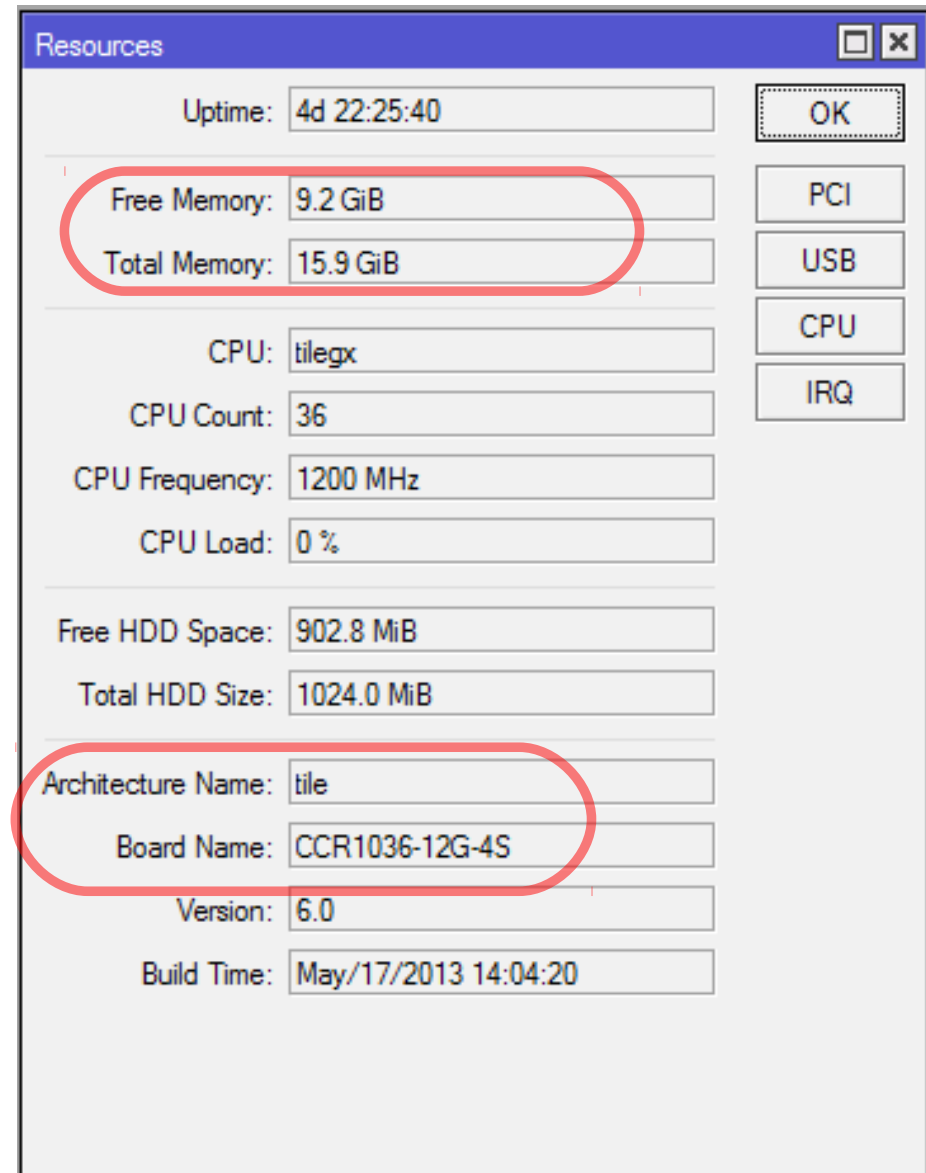
# New CPU architecture support

- In v5.x there were 4 different architectures
  - **mipsle** (RB1xx, RB5xx)
  - **mipsbe** (RB4xx, RB7xx, RB9xx, RB2011, SXT, Sextant, Groove, Metal, CRS)
  - **ppc** (RB1xxx, RB6xx, RB8xx)
  - **x86**
- In v6.x there will be one more
  - **tile** (CCR1xxx)



# RouterOS Tile architecture

- Only for CCR devices
- 64-bit operating system (more RAM)
- Dual memory channel support (faster RAM)
- Hardware Accelerated Multi-threading (no need for RPS and IRQ management)

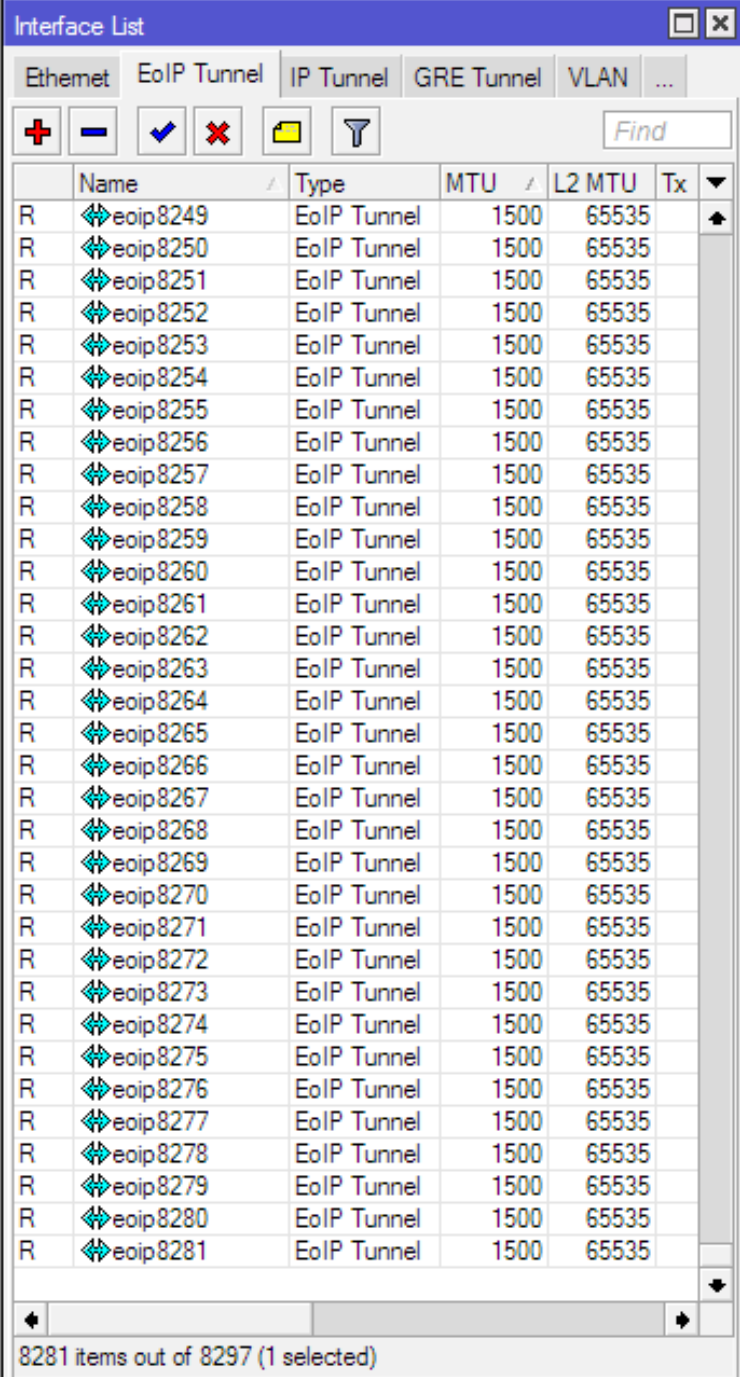


The screenshot shows the 'Resources' window in RouterOS. It displays various system metrics. Two red ovals highlight specific information: the memory status (Free Memory: 9.2 GiB, Total Memory: 15.9 GiB) and the hardware details (Architecture Name: tile, Board Name: CCR1036-12G-4S). On the right side, there are buttons for 'OK', 'PCI', 'USB', 'CPU', and 'IRQ'.

Resources	
Uptime:	4d 22:25:40
Free Memory:	9.2 GiB
Total Memory:	15.9 GiB
CPU:	tilegx
CPU Count:	36
CPU Frequency:	1200 MHz
CPU Load:	0 %
Free HDD Space:	902.8 MiB
Total HDD Size:	1024.0 MiB
Architecture Name:	tile
Board Name:	CCR1036-12G-4S
Version:	6.0
Build Time:	May/17/2013 14:04:20

# New Kernel Features

- Newest interface driver support for x86 systems
- Improved interface management - scales well for up to thousands of interfaces and more
- Uses less space on storage - works well with 32MiB flash



	Name	Type	MTU	L2 MTU	Tx
R	↔e0ip8249	EoIP Tunnel	1500	65535	
R	↔e0ip8250	EoIP Tunnel	1500	65535	
R	↔e0ip8251	EoIP Tunnel	1500	65535	
R	↔e0ip8252	EoIP Tunnel	1500	65535	
R	↔e0ip8253	EoIP Tunnel	1500	65535	
R	↔e0ip8254	EoIP Tunnel	1500	65535	
R	↔e0ip8255	EoIP Tunnel	1500	65535	
R	↔e0ip8256	EoIP Tunnel	1500	65535	
R	↔e0ip8257	EoIP Tunnel	1500	65535	
R	↔e0ip8258	EoIP Tunnel	1500	65535	
R	↔e0ip8259	EoIP Tunnel	1500	65535	
R	↔e0ip8260	EoIP Tunnel	1500	65535	
R	↔e0ip8261	EoIP Tunnel	1500	65535	
R	↔e0ip8262	EoIP Tunnel	1500	65535	
R	↔e0ip8263	EoIP Tunnel	1500	65535	
R	↔e0ip8264	EoIP Tunnel	1500	65535	
R	↔e0ip8265	EoIP Tunnel	1500	65535	
R	↔e0ip8266	EoIP Tunnel	1500	65535	
R	↔e0ip8267	EoIP Tunnel	1500	65535	
R	↔e0ip8268	EoIP Tunnel	1500	65535	
R	↔e0ip8269	EoIP Tunnel	1500	65535	
R	↔e0ip8270	EoIP Tunnel	1500	65535	
R	↔e0ip8271	EoIP Tunnel	1500	65535	
R	↔e0ip8272	EoIP Tunnel	1500	65535	
R	↔e0ip8273	EoIP Tunnel	1500	65535	
R	↔e0ip8274	EoIP Tunnel	1500	65535	
R	↔e0ip8275	EoIP Tunnel	1500	65535	
R	↔e0ip8276	EoIP Tunnel	1500	65535	
R	↔e0ip8277	EoIP Tunnel	1500	65535	
R	↔e0ip8278	EoIP Tunnel	1500	65535	
R	↔e0ip8279	EoIP Tunnel	1500	65535	
R	↔e0ip8280	EoIP Tunnel	1500	65535	
R	↔e0ip8281	EoIP Tunnel	1500	65535	

8281 items out of 8297 (1 selected)

CPU				
Find				
CPU	Load (%)	IRQ (%)	Disk (%)	
cpu0	0	0	0	
cpu1	0	0	0	
cpu2	0	0	0	
cpu3	0	0	0	
cpu4	0	0	0	
cpu5	0	0	0	
cpu6	0	0	0	
cpu7	0	0	0	
cpu8	0	0	0	
cpu9	0	0	0	
cpu10	0	0	0	
cpu11	0	0	0	
cpu12	0	0	0	
cpu13	0	0	0	
cpu14	0	0	0	
cpu15	0	0	0	
cpu16	0	0	0	
cpu17	0	0	0	
cpu18	0	0	0	
cpu19	0	0	0	
cpu20	0	0	0	
cpu21	0	0	0	
cpu22	0	0	0	
cpu23	0	0	0	
cpu24	0	0	0	
cpu25	0	0	0	
cpu26	0	0	0	
cpu27	0	0	0	
cpu28	0	0	0	
cpu29	0	0	0	
cpu30	0	0	0	
cpu31	0	0	0	
cpu32	0	0	0	
cpu33	0	0	0	
cpu34	0	0	0	
cpu35	0	0	0	

36 items

# What else is new?

- Lifted 16 CPU core limit
- Improved RouterOS performance on multi-cpu systems (up to 20%)
- Improved RouterBOARD interface driver performance (up to 30%)
- Routerboard package is now merged into system package

# Fast Path

- Fast Path allows to forward packets without additional processing in the Linux Kernel. It improves forwarding speeds significantly.
- Fast path requirements
  - Fast Path should be allowed in configuration
  - Interface driver must have support
  - Specific configuration conditions
- Currently RouterOS has fast path handlers for: ipv4 routing, traffic generator, mpls, bridge
- More handlers will be added in future.

# New Throughput test results

RB951G-2HnD		Gigabit Ethernet test (600Mhz)		RouterOS v6.0rc5			
Mode	Configuration	64 byte		512 byte		1518 byte	
		kpps	Mbps	kpps	Mbps	kpps	Mbps
Bridging	none (fast path)	269.6	176.9	232	983.7	81	995.3
Bridging	25 Bridge filter rules	87.6	57.5	86	364.6	81	995.3
Routing	none (fast path)	226.9	148.8	210	890.4	81	995.3
Routing	25 Simple Queues	226.9	148.8	210	890.4	81	995.3
Routing	25 IP filter rules	226.9	148.8	210	890.4	81	995.3

RB1100Hx2				RouterOS v6.0rc5			
Mode	Configuration	64 byte		512 byte		1518 byte	
		kpps	Mbps	kpps	Mbps	kpps	Mbps
Bridging	none (fast path)	1690	1108.6	704	2985.0	406	4988.9
Bridging	25 Bridge filter rules	412	270.3	396	1679.0	308	3784.7
Routing	none (fast path)	1495	990.7	704	2985.0	345	4220.4
Routing	25 Simple Queues	1495	990.7	704	2985.0	345	4220.4
Routing	25 IP filter rules	1495	990.7	704	2985.0	345	4220.4

CCR1036-12G-4S				RouterOS v6.0rc5			
Mode	Configuration	64 byte		512 byte		1518 byte	
		kpps	Mbps	kpps	Mbps	kpps	Mbps
Bridging	none (fast path)	23808	15618.0	3759	15938.2	1300	15974.4
Bridging	25 Bridge filter rules	7340	4815.0	3759	15938.2	1300	15974.4
Routing	none (fast path)	23808	15618.0	3759	15938.2	1300	15974.4
Routing	25 Simple Queues	7919	5194.9	3759	15938.2	1300	15974.4
Routing	25 IP filter rules	3127	2051.3	2998	12711.5	1300	15974.4

# Throughput in millions pps

Interface List								
Interface	Ethernet	EoIP Tunnel	IP Tunnel	GRE Tunnel	VLAN	VRRP	Bonding	LTE
✓	✗	📄	🔍					
	Name	Type	MTU	L2 MTU	Tx	Rx	Tx Packet...	Rx Packet (...)
RS	ether1	Ethernet	1500	1590	478.5 Mbps	465.8 Mbps	996 885	970 618
RS	ether2	Ethernet	1500	1590	477.2 Mbps	480.3 Mbps	994 356	1 000 701
RS	ether3	Ethernet	1500	1590	475.1 Mbps	513.4 Mbps	989 969	1 069 736
RS	ether4	Ethernet	1500	1590	476.6 Mbps	492.0 Mbps	993 024	1 025 024
RS	ether5	Ethernet	1500	1590	475.8 Mbps	501.4 Mbps	991 399	1 044 710
RS	ether6	Ethernet	1500	1590	478.4 Mbps	469.2 Mbps	996 816	977 502
RS	ether7	Ethernet	1500	1590	478.1 Mbps	471.7 Mbps	996 120	982 714
RS	ether8	Ethernet	1500	1590	482.7 Mbps	408.8 Mbps	1 005 632	851 693
RS	ether9	Ethernet	1500	1590	477.1 Mbps	487.0 Mbps	994 065	1 014 717
RS	ether10	Ethernet	1500	1590	478.2 Mbps	468.2 Mbps	996 343	975 495
RS	ether11	Ethernet	1500	1590	479.2 Mbps	455.3 Mbps	998 539	948 640
R	ether12	Ethernet	1500	1590				
RS	sfp1	Ethernet	1500	1590				
RS	sfp2	Ethernet	1500	1590				
RS	sfp3	Ethernet	1500	1590				
RS	sfp4	Ethernet	1500	1590				
16 items out of 17								

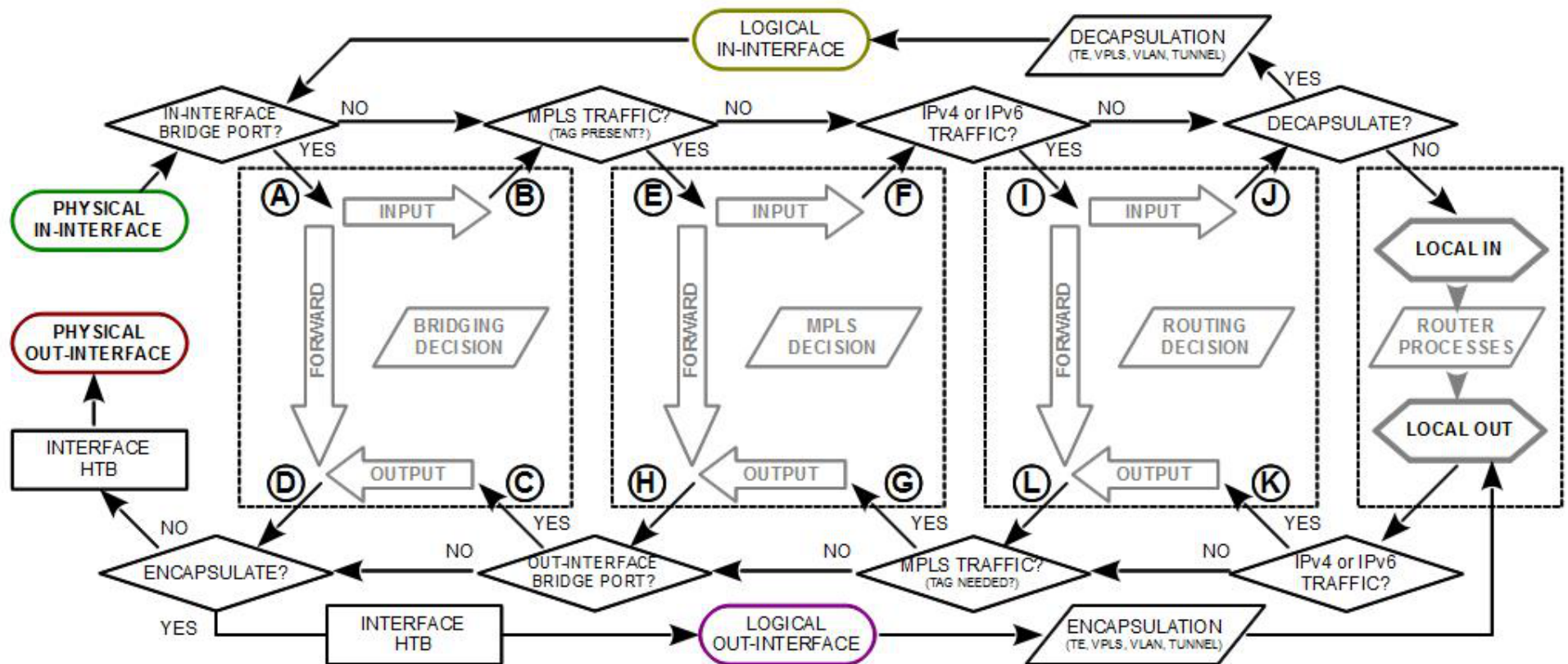
```
[admin@RouterOS] > interface monitor-traffic aggregate
rx-packets-per-second: 15 577 081
rx-drops-per-second: 0
rx-errors-per-second: 0
rx-bits-per-second: 7.4Gbps
tx-packets-per-second: 15 576 803
tx-drops-per-second: 0
tx-errors-per-second: 0
tx-bits-per-second: 7.4Gbps
-- [Q quit|D dump|C-z pause]
```

# Traffic Generator Tool

- Traffic Generator is a bandwidth-tool evolution
- Traffic Generator can:
  - Determine transfer rates, packet loss
  - Detect out-of-order packets
  - Collect latency and jitter values
  - Inject and replay \*.pcap file (NEW!! in v6.1)
  - Working on TCP protocol emulation
- “Quick” mode
- Full Winbox support (coming soon)



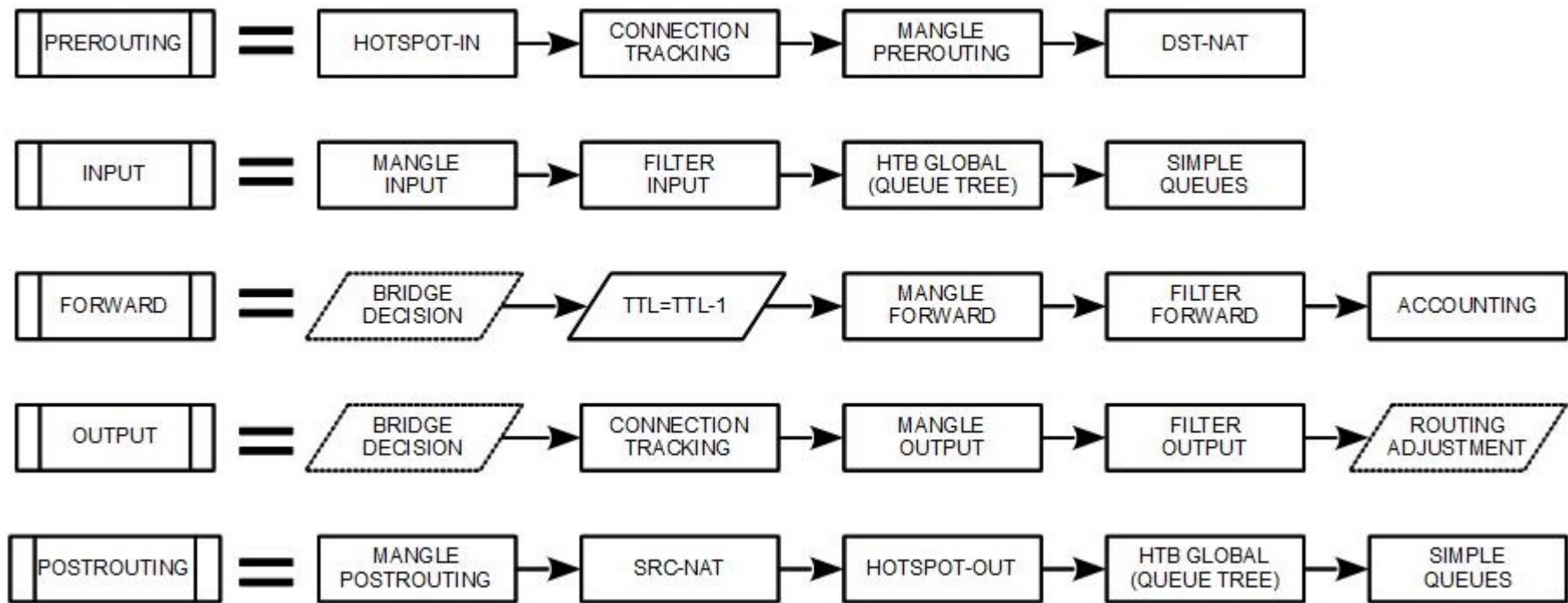
# MikroTik RouterOS Packet Flow Diagram for version 6.x







# Yes, still - Packet Flow Diagram (page 3)

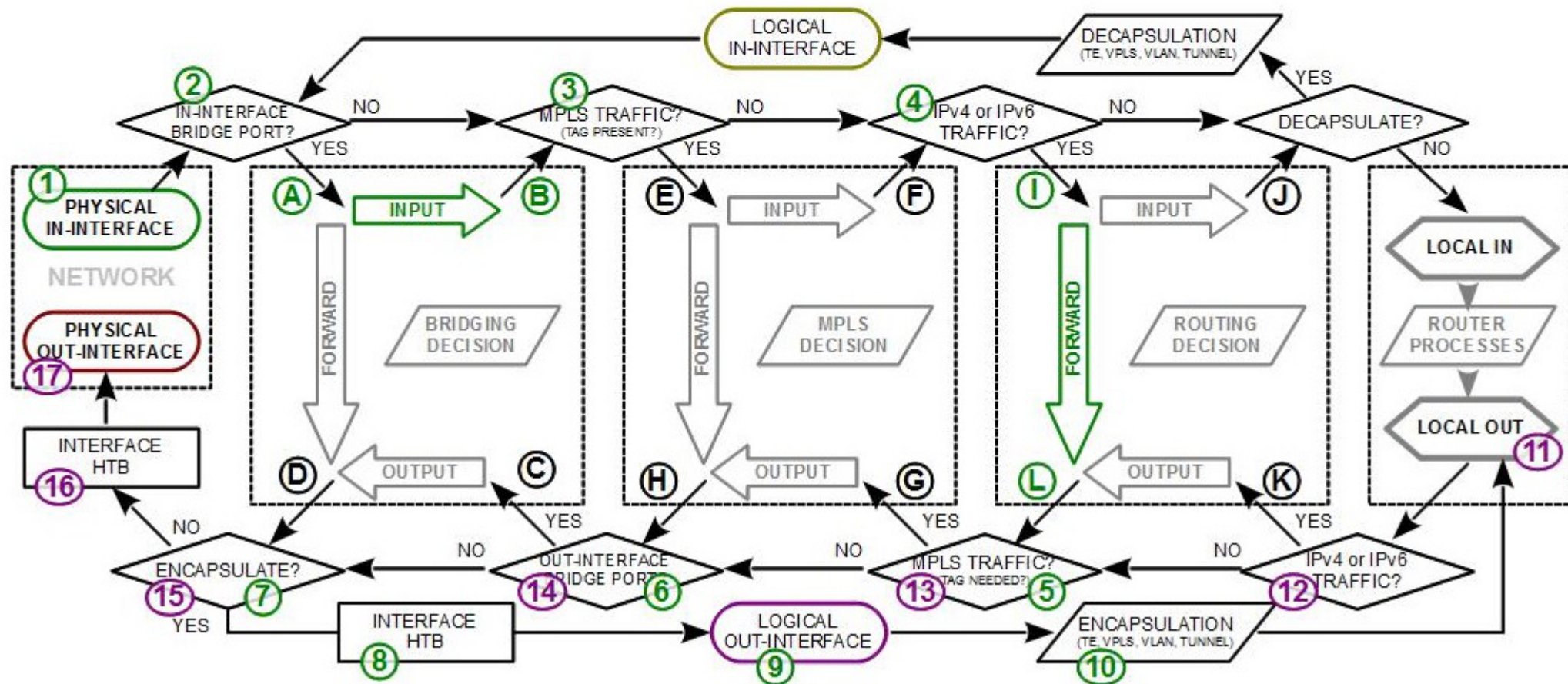


```

graph LR
    subgraph ROUTER
        bridge1
        ether1
        port_ether2_in([port: ether2-in])
        port_wlan1([port: wlan1])
        pppoe_out([pppoe-out])
        bridge1 --- port_ether2_in
        bridge1 --- port_wlan1
        bridge1 --- ether1
        ether1 --- pppoe_out
    end
    In(( )) -- green arrow --> bridge1
    pppoe_out -- purple arrow --> Out(( ))

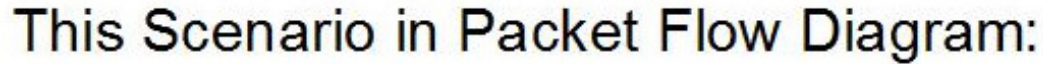
```

### This Scenario in Packet Flow Diagram:

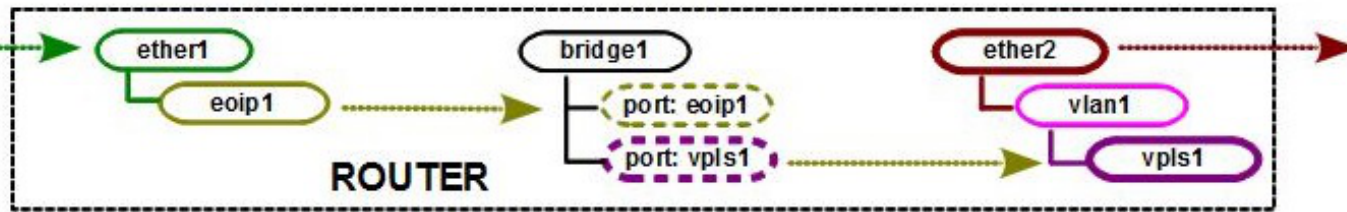




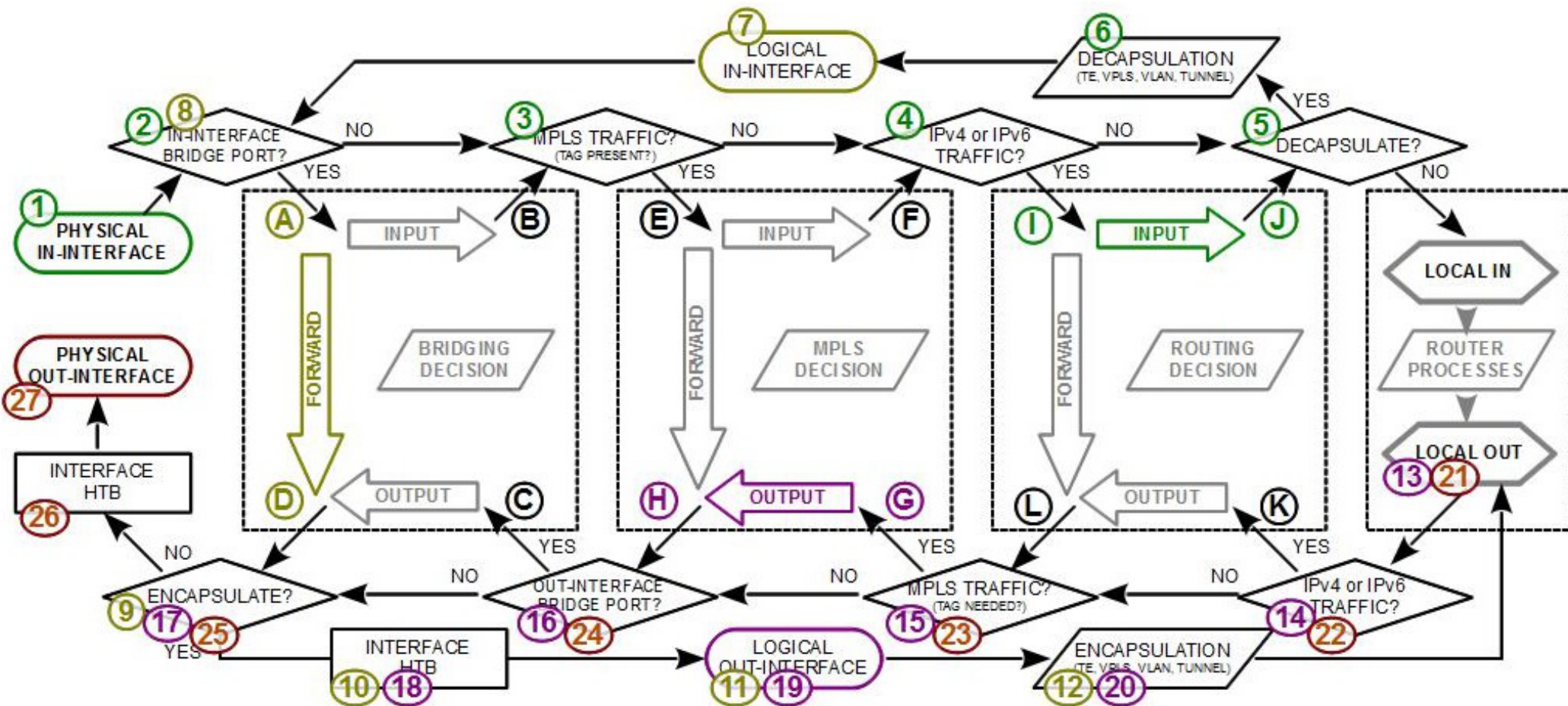
## ROUTER



# Packet Flow Scenario:



## This Scenario in Packet Flow Diagram:



# QoS System Reworked

- **WARNING!!!**

Simple Queues and Queue Tree queues in some specific configurations might be inactivated after upgrade from RouterOS v3.x, v4.x, v5.x to RouterOS v6.x

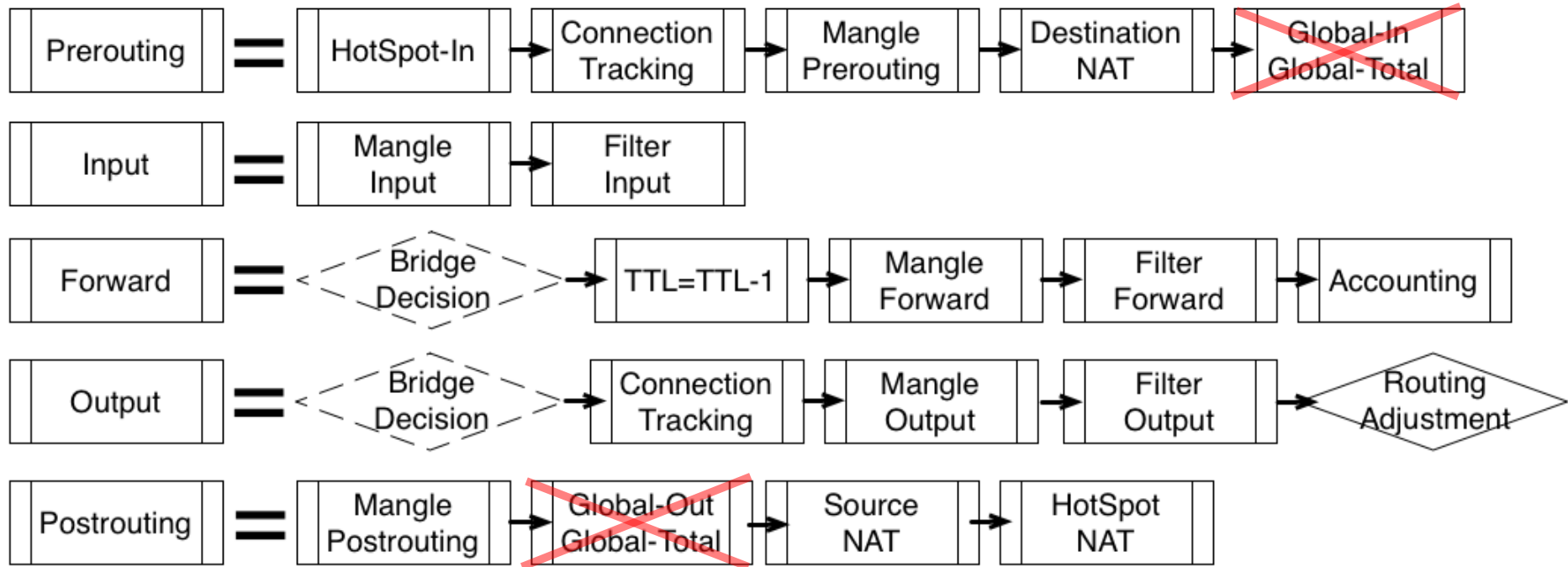
Automatic configuration transition is unavailable cause in some configurations might result in inability to access router.

# Changes in Packet Flow Diagram

- Queuing packet in several different places in its “life-cycle” causes enormous performance degradation in multi-threading environment
- In RouterOS v6.x QoS system was redesigned so that queuing happens in the same place respectively to other processes in the router.
- HTB “global-in”, “global-out” and “global-total” was deleted, and replaced with HTB “global” and placed at the very end of packet's “life-cycle” in the router.

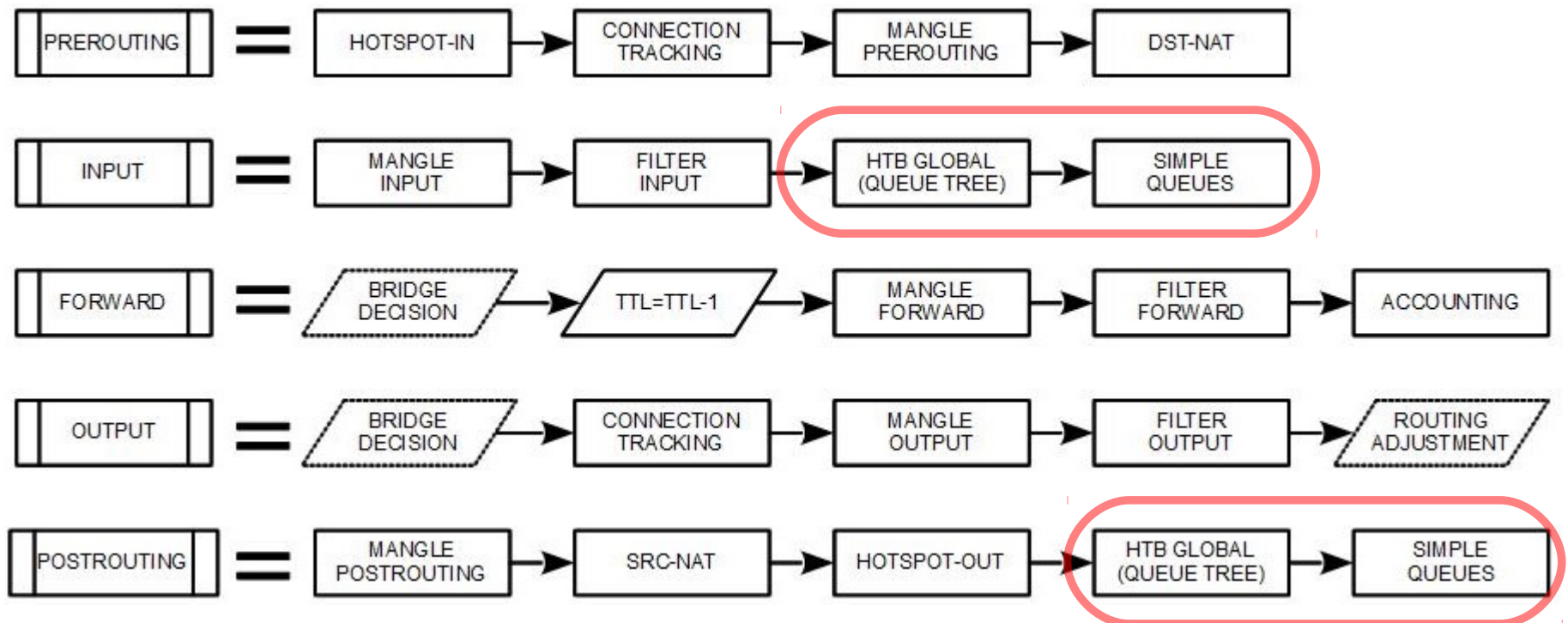


# HTB in RouterOS v5





# HTB in RouterOS v6



# Additional changes

- Simple queues are now separated from the Queue Tree completely – the same packet can be captured in HTB “global” and Simple Queues (in v5 Simple Queues shared global-HTBs)
- As all queuing now happens after SRC-NAT the PCQ queue type is updated and now is NAT aware (from connection tracking)
- You can now specify multiple packet-marks per single queue

# Simple Queues

- Simple queue matching algorithm has been updated
- Very small overhead for packets that miss existing simple queues
- Top level simple queues are now balanced between CPU cores (32 queues 9x faster than 1 queue on CCR1036)

Queue List				
Simple Queues				
Interface Queues				
Queue Tree				
Queue Types				
+ - ✓ ✗ [icon] [icon] [icon] Reset Counters 00 Reset A				
#	Name	Target	Rx Max Limit	Tx Max Limit
24967	queue24968	4.4.100.218	1M	1M
24968	queue24969	4.4.100.219	1M	1M
24969	queue24970	4.4.100.220	1M	1M
24970	queue24971	4.4.100.221	1M	1M
24971	queue24972	4.4.100.222	1M	1M
24972	queue24973	4.4.100.223	1M	1M
24973	queue24974	4.4.100.224	1M	1M
24974	queue24975	4.4.100.225	1M	1M
24975	queue24976	4.4.100.226	1M	1M
24976	queue24977	4.4.100.227	1M	1M
24977	queue24978	4.4.100.228	1M	1M
24978	queue24979	4.4.100.229	1M	1M
24979	queue24980	4.4.100.230	1M	1M
24980	queue24981	4.4.100.231	1M	1M
24981	queue24982	4.4.100.232	1M	1M
24982	queue24983	4.4.100.233	1M	1M
24983	queue24984	4.4.100.234	1M	1M
24984	queue24985	4.4.100.235	1M	1M
24985	queue24986	4.4.100.236	1M	1M
24986	queue24987	4.4.100.237	1M	1M
24987	queue24988	4.4.100.238	1M	1M
24988	queue24989	4.4.100.239	1M	1M
24989	queue24990	4.4.100.240	1M	1M
24990	queue24991	4.4.100.241	1M	1M
24991	queue24992	4.4.100.242	1M	1M
24992	queue24993	4.4.100.243	1M	1M
24993	queue24994	4.4.100.244	1M	1M
24994	queue24995	4.4.100.245	1M	1M
24995	queue24996	4.4.100.246	1M	1M
24996	queue24997	4.4.100.247	1M	1M
24997	queue24998	4.4.100.248	1M	1M
24998	queue24999	4.4.100.249	1M	1M
24999	queue25000	4.4.100.250	1M	1M
25000 items		0 B queued		0 packets queued

# Simpler Simple Queues

- “target-addresses” and “interface” parameters are joined into one “target” parameter
- “dst-address” parameter is changed to “dst” and now can be specified as an interface as well
- direction and p2p parameters removed
- “target” now must be specified on a simple queue creation
- Separate “priority” parameter for download, upload and total

# Simple Queue Interface v5

The image displays two side-by-side screenshots of the 'New Simple Queue' configuration window, showing different tabs.

**Left Window (General Tab):**

- Name:** queue\_from\_v5
- Target Address:** 192.168.1.254
- Target Upload:** ☒ **Target Download:** ☒
- Max Limit:** 20M (Upload) / 20M (Download) bits/s
- Burst:**
  - Burst Limit:** unlimited (Upload) / unlimited (Download) bits/s
  - Burst Threshold:** unlimited (Upload) / unlimited (Download) bits/s
  - Burst Time:** 0 (Upload) / 0 (Download) s
- Time:** 00:00:00 - 1d 00:00:00
- Days:** ☒ sun ☒ mon ☒ tue ☒ wed ☒ thu ☒ fri ☒ sat
- enabled** (checkbox)

**Right Window (Advanced Tab):**

- P2P:** (dropdown menu)
- Packet Marks:** (text field)
- Dest. Address:** (text field)
- Interface:** all (dropdown menu)
- Limit At:** 2M (Upload) / 2M (Download) bits/s
- Queue Type:** pcq-upload-default (Upload) / pcq-download-default (Download) (dropdown menu)
- Parent:** none (dropdown menu)
- Priority:** 8 (text field)
- Buttons:** OK, Cancel, Apply, Disable, Comment, Copy, Remove, Reset Counters, Reset All Counters, Torch
- enabled** (checkbox)

# Simple Queue Interface v6

The image displays two side-by-side screenshots of the 'New Simple Queue' configuration window, showing different tabs and settings.

**Left Screenshot (General Tab):**

- Name:** queue\_from\_v6
- Target:** 192.168.1.254 (highlighted with a red circle)
- Dst.:** ether7 (highlighted with a red circle)
- Target Upload:** 20M
- Target Download:** 20M
- Limit At:** 2M
- Burst Limit:** unlimited
- Burst Threshold:** unlimited
- Burst Time:** 0
- Time:** 00:00:00 - 1d 00:00:00
- Days:** sun, mon, tue, wed, thu, fri, sat (all checked)
- enabled** (checkbox)

**Right Screenshot (General Tab):**

- Packet Marks:** (empty)
- Target Upload:** 2M
- Target Download:** 2M
- Limit At:** 2M
- Priority:** 6 (highlighted with a red circle) and 7 (highlighted with a red circle)
- Queue Type:** pcq-upload-default
- Parent:** none
- OK**, **Cancel**, **Apply**, **Disable**, **Comment**, **Copy**, **Remove**, **Reset Counters**, **Reset All Counters**, **Torch** (buttons)
- enabled** (checkbox)

# Changes in the Firewall

- Firewall now has “all-ether”, “all-wireless”, “all-vlan”, “all-ppp” as possibilities in interface matching
- New priority matcher
- New “change-dscp” options “from-priority” and “from priority-to-high-3-bits”
- New Mangle Actions “snif-tzsp”, “snif-pc” to send packet stream to remote sniffer.

# Changes in Dynamic stuff

- Only 2 dynamic “change-mss” mangle rules are created for “all-ppp” interfaces
- In PPTP, L2TP, SSTP, HotSpot profiles you can specify:
  - dynamic simple queue position (also true for DHCP leases)
  - dynamic simple queue parent
  - dynamic simple queue type
- PPP,PPTP,L2TP,SSTP - added default-route-distance parameter



# Changes in Tunnels

- PPTP, L2TP, SSTP – allows you to specify server via DNS name
- SSTP can now forces AES encryption instead of default RC4
- PPP profile now has a “bridge-path-cost” and “bridge-port-priority” parameters
- PPPoE server - allow service with empty service-name to accept all pppoe clients
- Hotspot and PPP now support multiple address-lists from RADIUS

# Changes in DHCP

- dhcp-options now can be specified by mixing different data types
- dhcp-client have custom dhcp-option feature (examples at: `/ip dhcp-client option print`)
- DHCPv4 client now have special-classless option for add-default-route parameter
- Possibility to add DHCP relay agent information option (Option 82)
- DHCPv6 DNS option support

# Changes in scripts

- `':global'`, `':local'` and `':set'` commands have new parameter `'do'` that allows assigning a block of commands to the variable;
- Added `"on-error"` argument to `':do'` command that is executed if the command raises an error
- Added a new `':return'` command that interrupts execution of script and passes argument as return value if script was called as function
- Added `'verbose'` argument to `'/import'` command that enables line-by-line script import

# Other Changes

- Slave flag now will show up for interfaces that are in bridge, bonding or switch group
- “/export compact” now is as default for “/export”, use “/export verbose” to get previous behavior
- Connected routes become inactive when Interface goes down.
- Configurable Linux Kernel options in /ip settings and /ipv6 settings menus
- Initial OpenFlow support

# IPSec Road Warrior

- RoadWarrior setups are now supported
  - Mode Conf support
  - passive IPSec peer mode
  - Xauth support ( xauth PSK and Hybrid RSA)
  - Policy templates and generate-policy
  - Peer groups
  - Multiple peers with the same IP can be used.
  - and more...

# SCEP Protocol Support

- Simple Certificate Enrollment protocol (SCEP)
- This protocol allows to:
  - get CA certificate from CA server or RA
  - create self-signed certificate with temp key
  - send certificate request to the server
  - protect CA operations with CA passphrase
- More info at:  
<http://wiki.mikrotik.com/wiki/Manual:System/Certificates#SCEP>

# Wireless Advanced Channels

- Works only between Atheros AR92xx chips and only between MikroTik devices
  - center frequency range:
    - 2192-2734mhz
    - 4800-6100mhz
- Choose precise center frequency (0.5MHz step)
- Choose channel width (2.5-30MHz, 0.5MHz step)
- Super-channel license is required to use advanced channels – it is free of charge (only signed document required about proper usage)

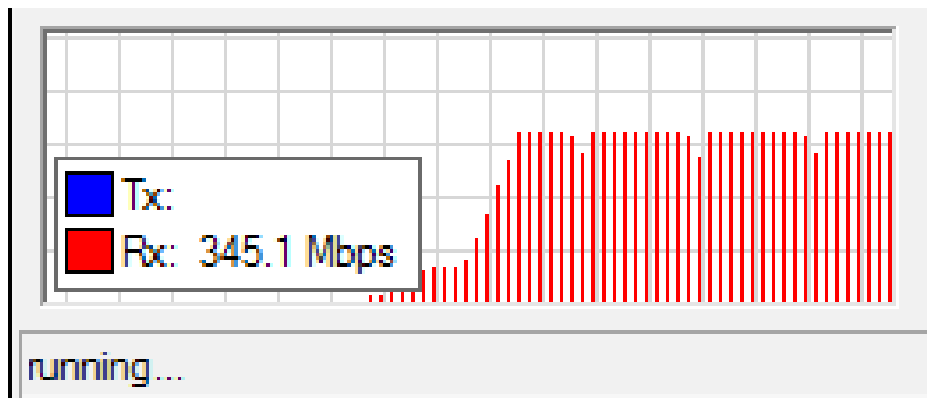
# Wireless Advanced Channels

- Located in /interface wireless channels
- Allows to name each advanced channel and group them into custom lists
- These names and list names later should be use in wireless clients scan-list, to enable them to see advanced channel APs (old style scan-list entries will not work)
- Custom scan-list options:
  - default, frequency, frequency range
  - advanced channel name or list name



# Advanced Channel Test

Current Tx Power	Status	Advanced Status	Traffic	...
Band: 5GHz-N				
Frequency: 5362.5MHz				
Wireless Protocol: 802.11				
Tx/Rx Rate: 27.0Mbps/405.0Mbps				



- Center frequency - 5362.5MHz
- Channel width - 30MHz
- Extension channel – Above
- Maximal data rate – MCS-15, 405Mbps
- Wireless protocol – 802.11n

# Questions!!!