



# Quality of Service (QoS) in VoIP networks

Mangle, Queue Tree

Aris Vink

Presales consultant, MikroTik trainer at SDC B.V.



©2018 Cluflor

25  
JAAR



# Who is SDC B.V. ?

- Distributor in ICT components
- Founded in 1992, located in Doetinchem(NL)
- Family company
- Supportive with solutions and service
- ICT partnerchannel in the Netherlands



Marketing&sales



Logistics



Academy



Professional services



Gigaset

NETGEAR



MikroTik



plantronics



SWOH



mum  
Mikrotik User Meeting

SDC

# Quality of Service in VoIP networks

- Quality of Service(QoS) introduction
- Circuit switched vs Packet switched networks
- QoS criteria for Voice over IP
- Problems in VoIP environments
- QoS traffic control
- Hierarchical Token Bucket (HTB)
- Qualify VoIP traffic with Mangle
- HTB implementation Queue Tree

# Quality of Service(QoS)

Quality of Service(QoS) is a general term that describes the measurement of the overall performance of a service such as telephony, video, file transfer or other network services.

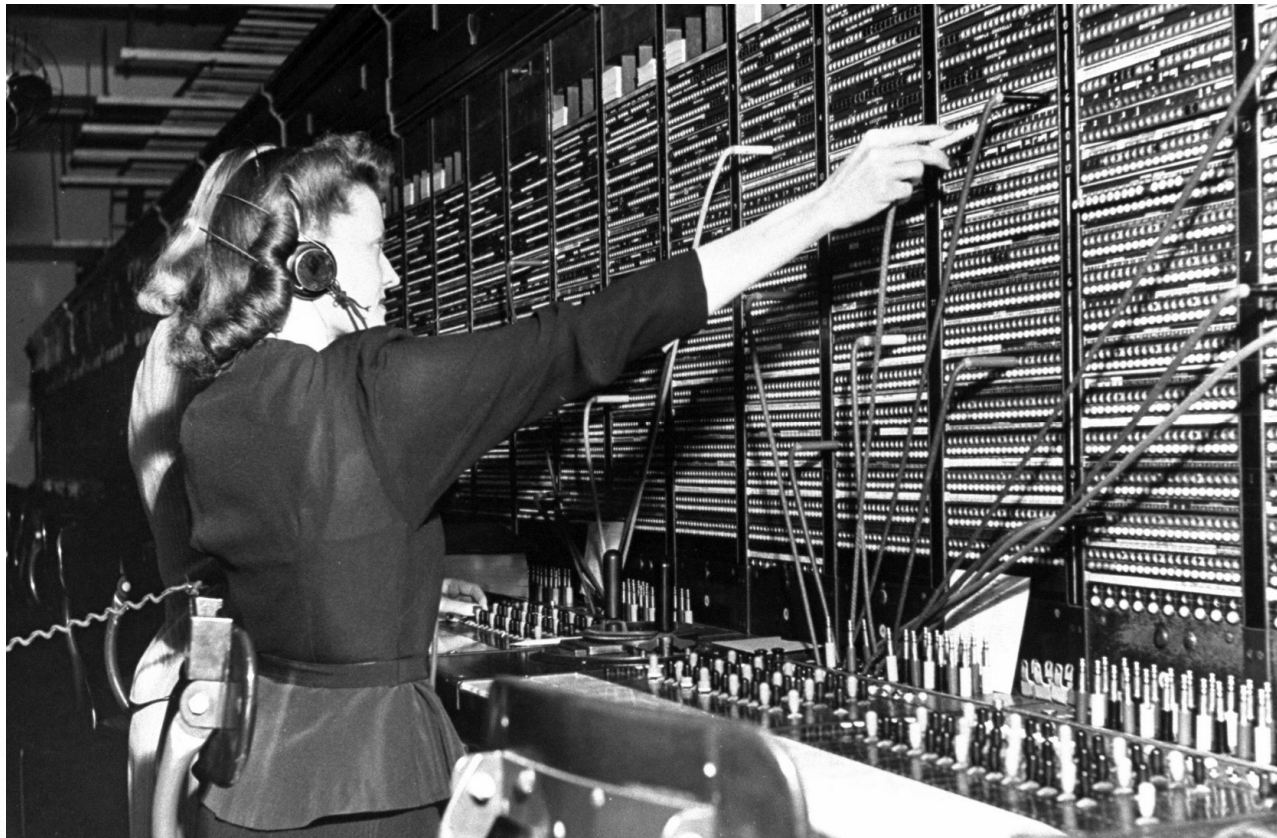
In the field of **computer networking**, quality of service refers to **traffic prioritization** and **resource reservation** control mechanisms. QoS is needed in order to **guarantee** a certain level of **performance** to a data flow

# Why the need for Quality of Service(QoS)

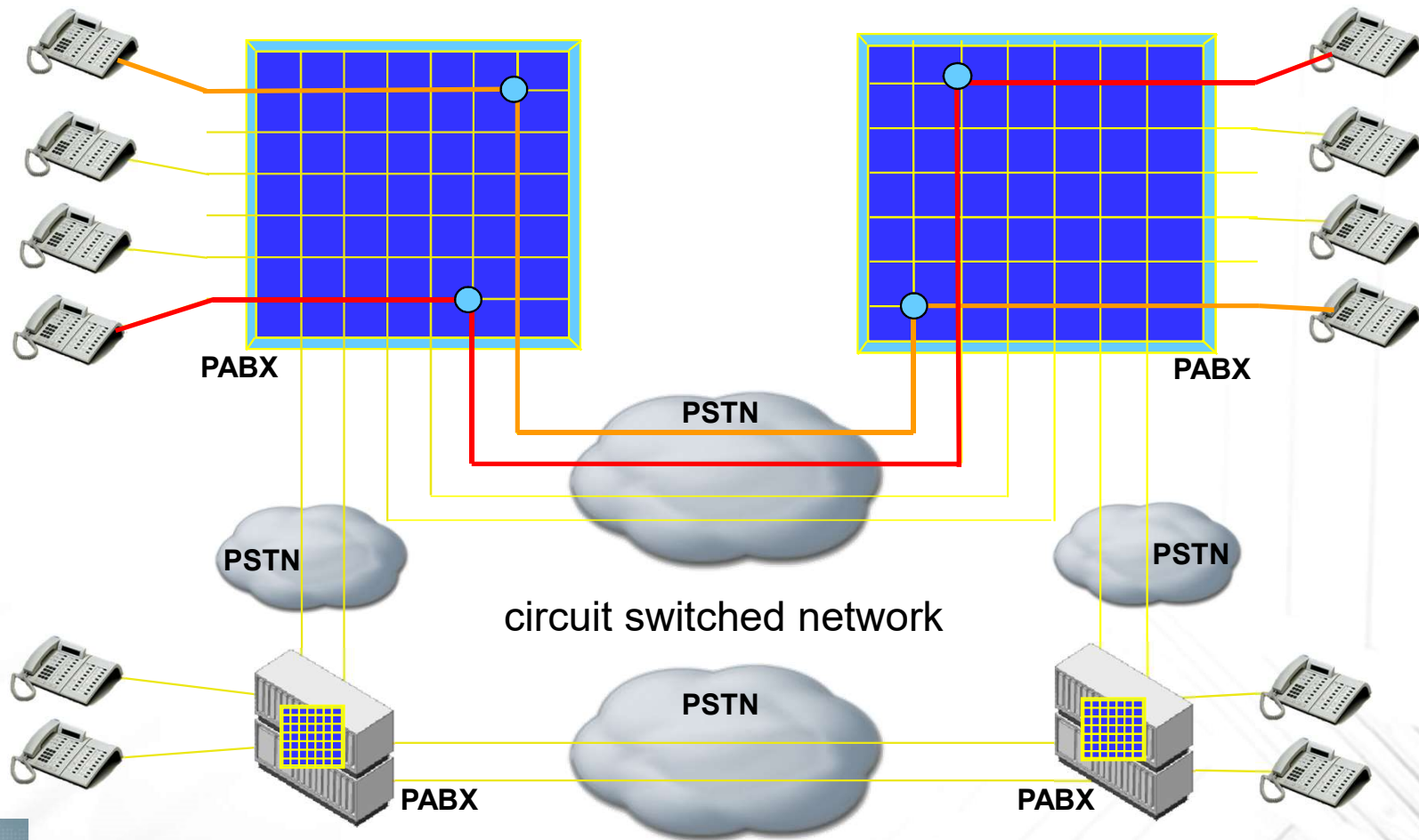
For instance Quality of Service is necessary to guarantee the bandwidth of an IP-phone. A phone call does not need much bandwidth but must always be instantly available for phone calls!



# Telephony history



# Circuit switched network (64kbit PCM telephony)

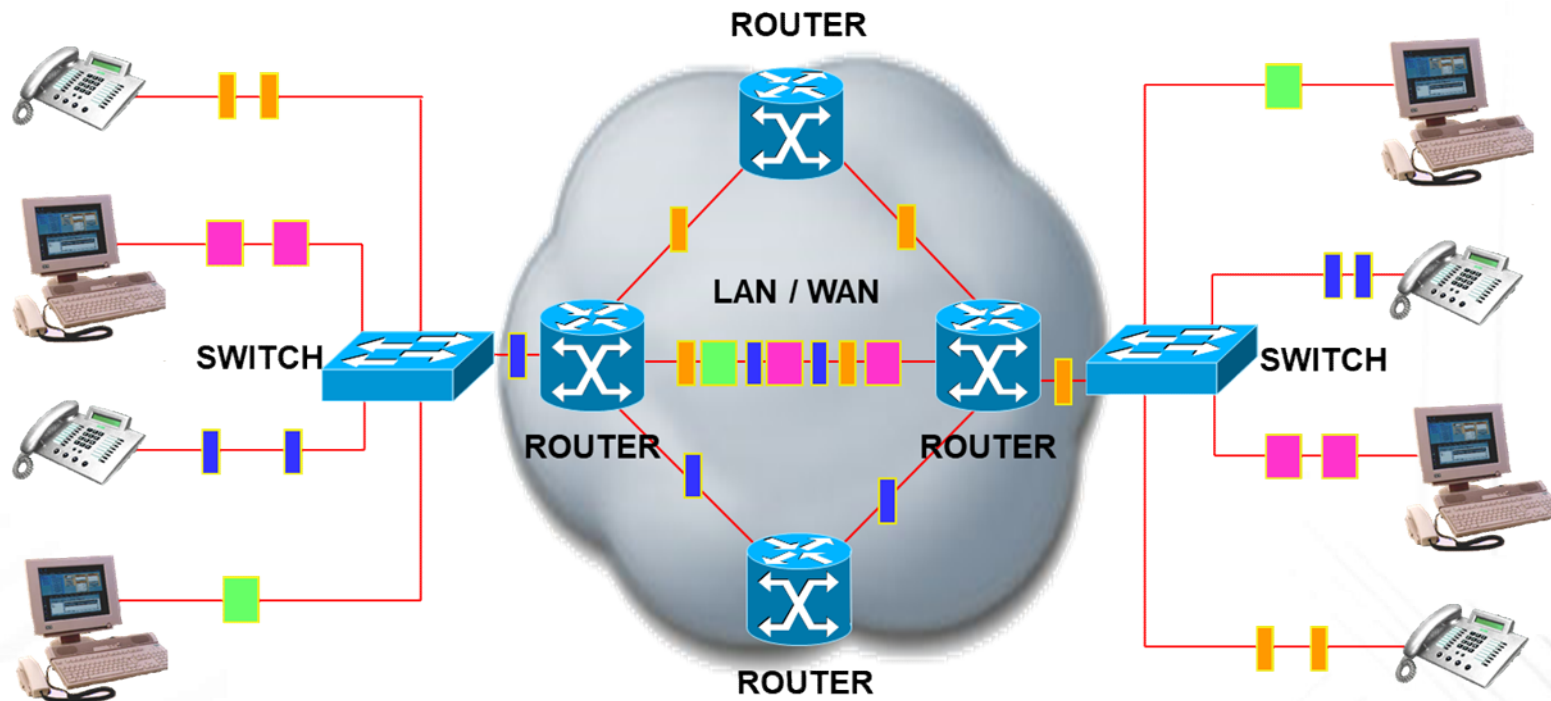




# Circuit switched network

- A **direct(fysical) connection** is made between phones. For setting up the circuit(call setup) a **seperate signalling network** is nessesary.
- A circuit switched network has a fixed(short) latency and for every session a fixed bandwidth. Circuit switched networks by standard have **QoS**.
- Unused bandwith within a circuit is unavailable for other services. Circuit switched networks are **unefficient** in this regard.

# Packet switched network



# Packet switched network

- IP phones exchange voice data over a **shared network**
- Since there is **no real connection**(circuit) the voice packets are **coded(IP)** so they can be routed through the network to the destination
- To **guarantee QoS** for telephony traffic identification, shaping en prioritizing has to be performed end to end.
- Services like speech, video and data share the same fysical network enabling **more efficient** use of network **bandwith**.

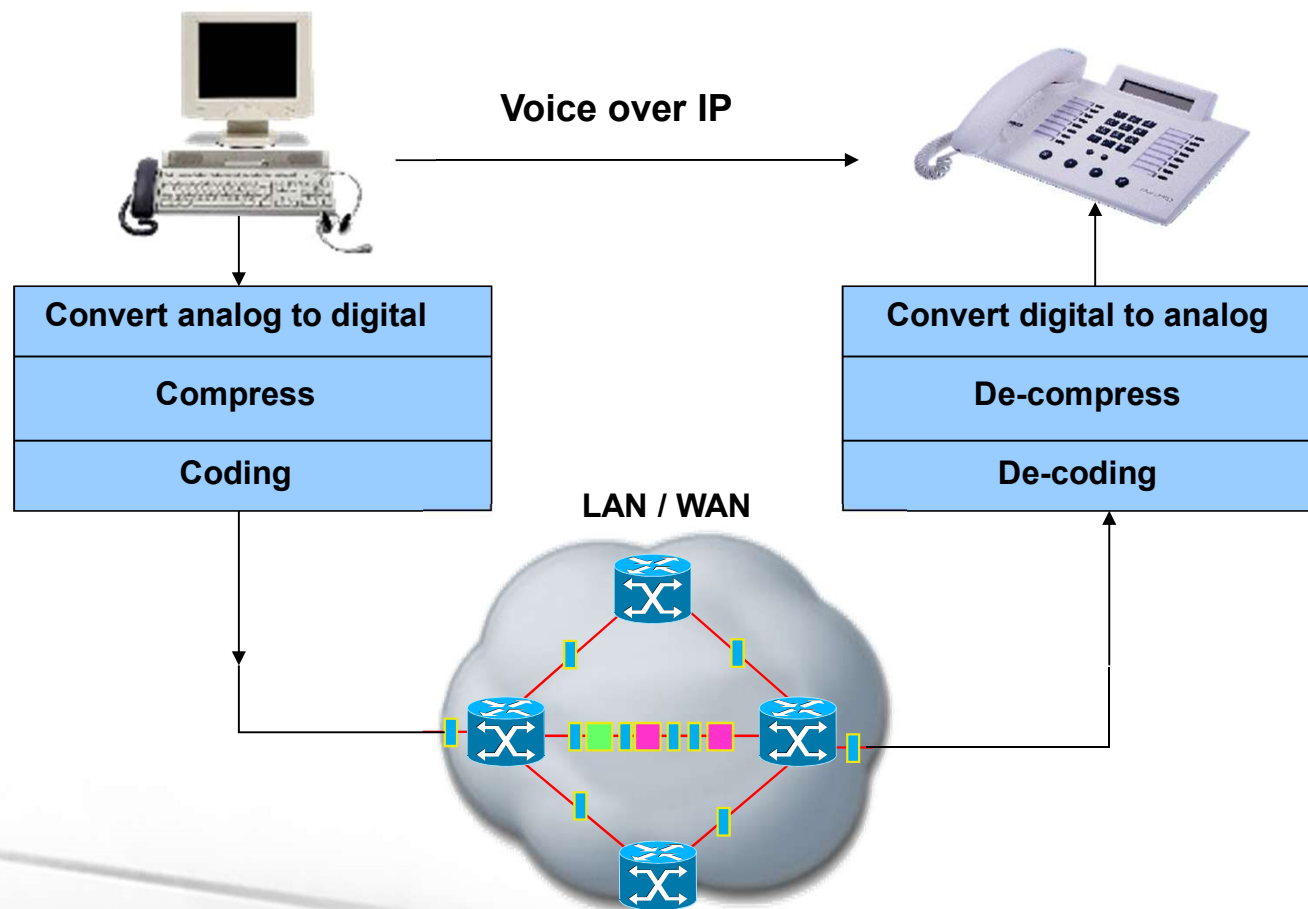
# QoS criteria for Voice over IP

- Sufficient bandwidth
- Availability of bandwidth(at all times)
- Low packet loss
- Low round trip delay(low latency)
- Minimal Jitter

## Problems in VoIP environments

- The quality of a VoIP call is heavily dependent on the network environment.
- **Latency, Jitter, & Packet Loss** can cause quality of experience issues for **Voice over IP (VoIP)** phone calls
- **Packet loss** is very common in IP networks, but certain networks such as WiFi can be particularly prone to high levels of packet loss.

# Bandwith usage



# Bandwidth usage

Bandwidth usages in SIP for						
code	Bandwidth for speech	+	Bandwidth for "IP" - Overhead	=	one cannel	bi-directional connection
G.711	64 kBit/s		ca. 19 kBit/s		ca. 83 kBit/s	ca. 166 kBit/s
G.723.1	6,3 kBit/s		ca. 19 kBit/s		ca. 25 kBit/s	ca. 50 kBit/s

# RTP(Realtime Transport Protocol)

SIP uses RTP for the end to end transport of audio and video data(**payload**). RTP adds extra information to the UDP header and in some cases can replace the UDP header.

- Sequence number
- Timestamp
- Payload type

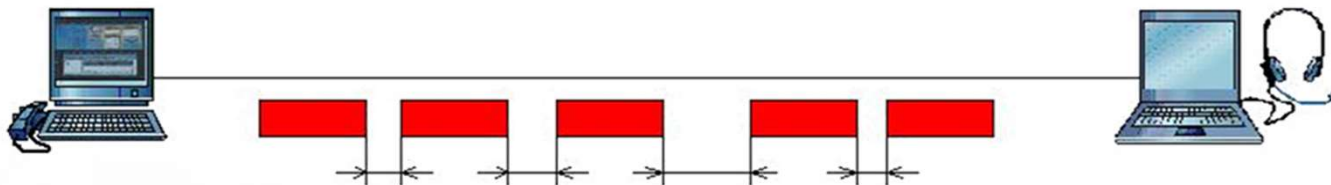


# RTCP(Realtime Transport Control Protocol)

RTP functionality is expanded by Real Time Control Protocol (RTCP). The goal of RTCP is to provide feedback to the VoIP endpoints about the quality of the data transmission. RTCP regularly sends reports with receipt statistics.

These statistics contain packet loss and jitter.

VoIP endpoints use this information to adapt the jitter buffer and/or the used audiocodec.



# Latency

- **Latency** is the time it takes the RTP (media) packets to traverse the network. Too much latency causes callers to speak over the top of each other.
- Callers start to notice the effect of **latency** around 250ms, above ~600ms the experience is unusable. There will always be some latency, the objective is to minimize it and keep total roundtrip time below 250ms.
- Ideally latency should be below 100ms because, while it is noticeable at 250ms, other services and issues beyond your control might add delay causing the cumulative total to be over 250ms.

# Jitter

- **Jitter** is when the latency through the network is not fixed but varies. Jitter causes that packets don't arrive in the same order they were sent. For small amounts of jitter, this can be resolved in the **jitter**buffer. The length of the jitter buffer introduced causes increased latency.
- Too much jitter cannot be resolved by a reasonable length jitter buffer without introducing too much delay, so instead results in **jitter induced packet loss** causing choppy audio.

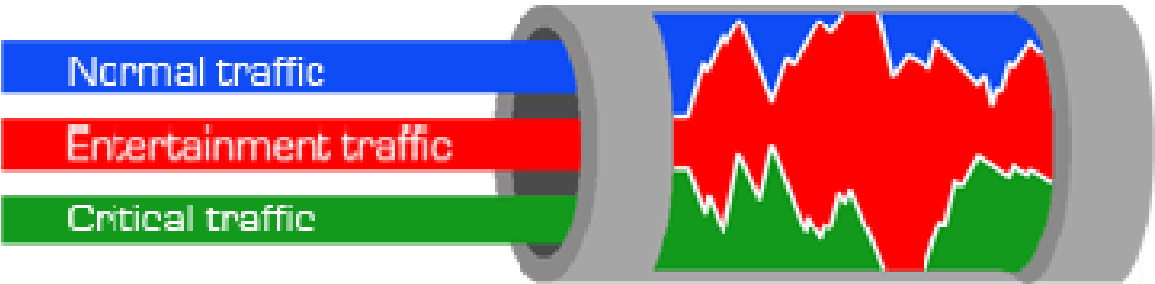
# QoS traffic control



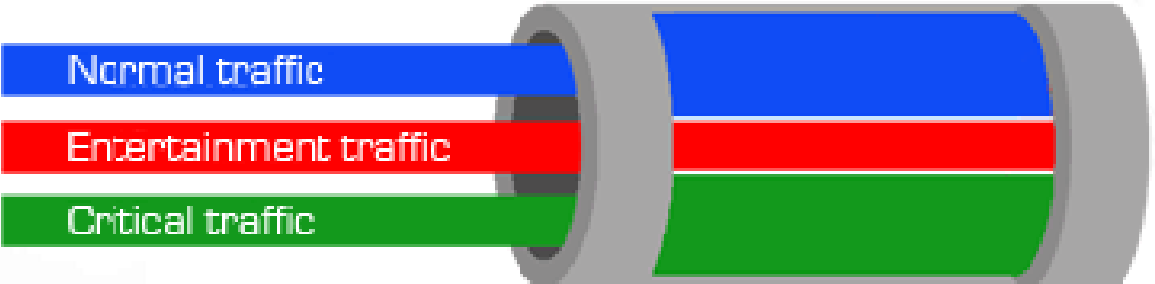
- Traffic shaping
- Prioritizing

# QoS traffic shaping.

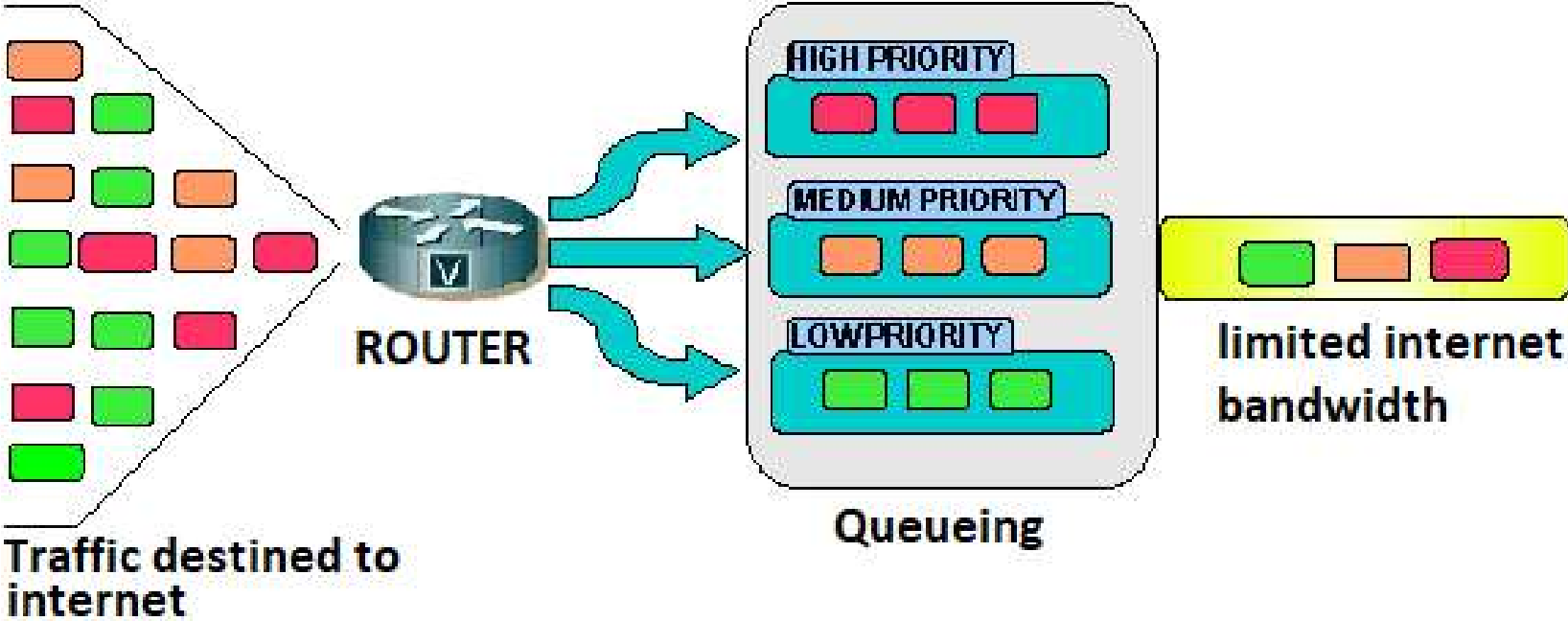
Bandwidth Use without QoS control



Bandwidth Use with QoS control



# QoS traffic prioritizing



## QoS traffic control

Traffic control is done on the **outbound interface** (we have no direct control over traffic that is being sent to us)

- **Rate limiting** 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 the packets! The packets will leave the router in the **exact sequence** as they are received (provided that we are forwarding them)

# Hierarchical Token Bucket (HTB)

Hierarchical Token Bucket(HTB) is a classful queuing method for handling different kinds of traffic.

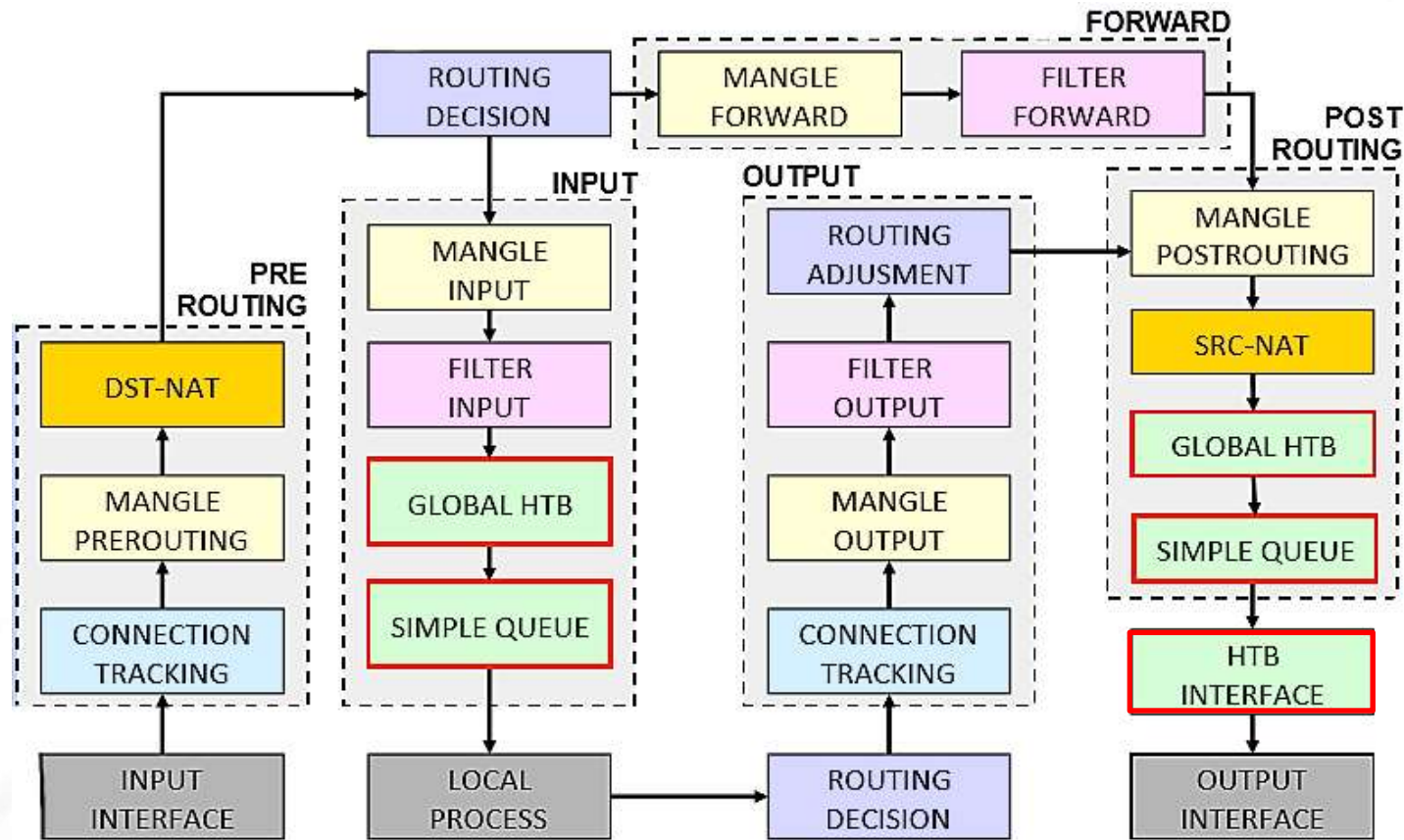
- All QoS implementation in **RouterOS** is based on Hierarchical Token Bucket.
- HTB provides a way to create a hierarchical queue structure and determine relations between parent and child queues and relations between child queues.



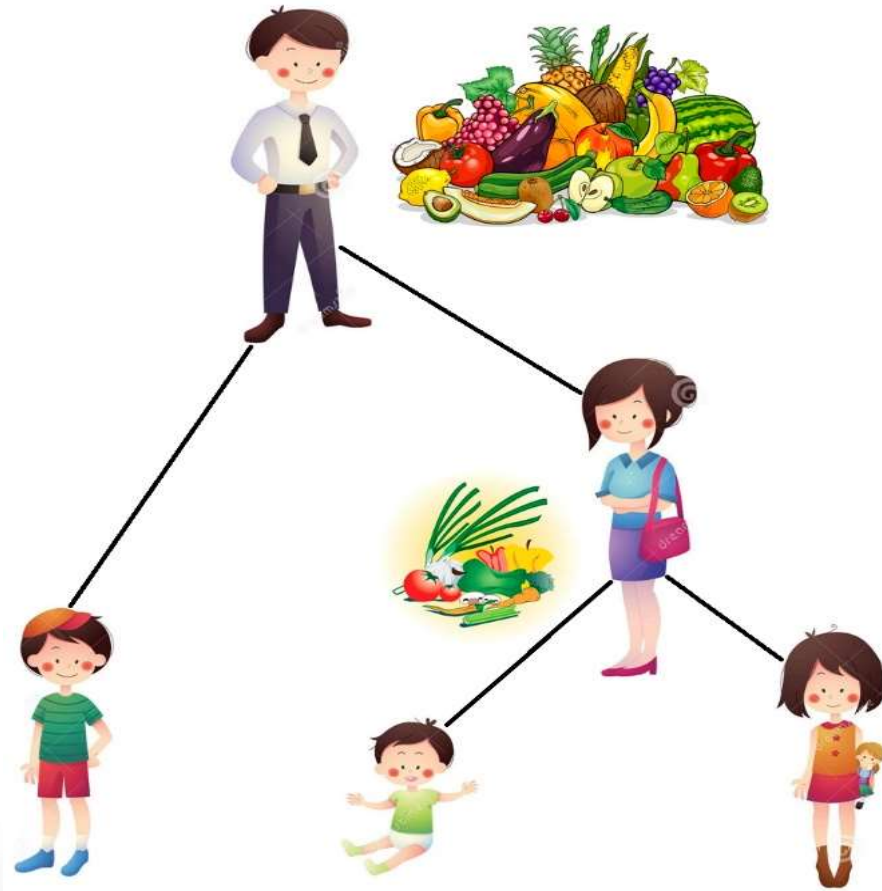
# Hierarchical Token Bucket (HTB)

- When a packet travels through the router, it passes **2 HTB trees (global and interface)** and simple queues
- When a packet travels to the router, it passes only **global HTB** and simple queues.
- When a packet travels from the router, it passes **2 HTB trees** and simple queues.

# HTB packet flow routerOS V6



# HTB queue structure(children and parents)



## HTB queue structure (children and parents)

- As soon as queue has at least one child it becomes a parent queue.
- All child queues (it doesn't matter how many levels of parents they have) are on the same bottom level of HTB
- Child queues do the actual traffic consumption, parent queues are only responsible for traffic distribution

# HTB rate limiting

HTB has two rate limits:

- **CIR (Committed Information Rate)** – (**limit-at** in RouterOS) worst case scenario, the flow will get this amount of traffic no matter what (assuming there is enough bandwidth)
- **MIR (Maximal Information Rate)** – (**max-limit** in RouterOS) best case scenario, rate that the flow can get up to, if the queue's parent has spare bandwidth available

At first HTB will try to satisfy every child queue's **limit-at** only then it will try to reach max-limit

## HTB rate limiting

Maximal rate of the parent should be equal or bigger than sum of committed rates of the children

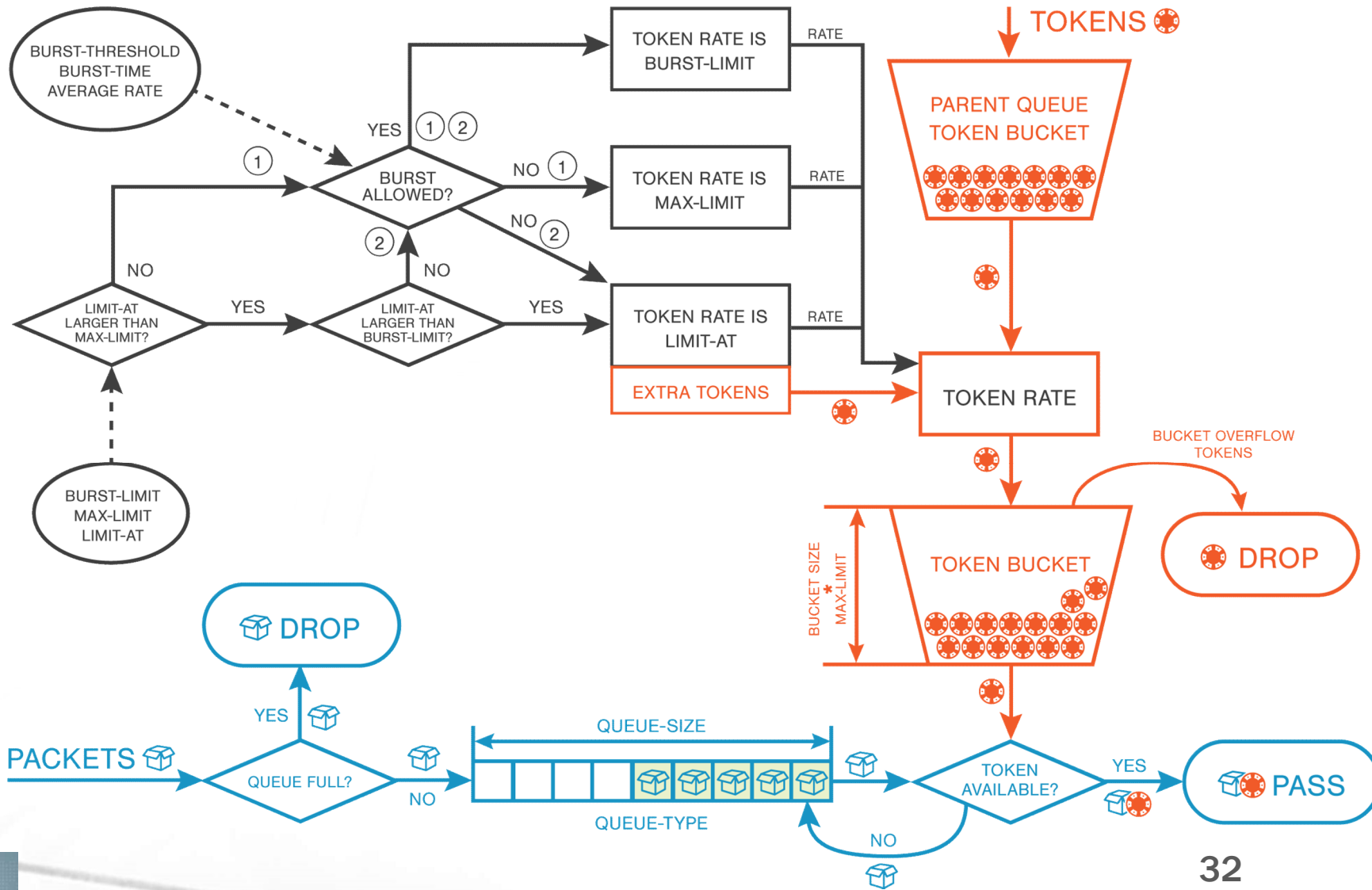
- $MIR(\text{parent}) \geq CIR(\text{child1}) + \dots + CIR(\text{childN})$

Maximal rate of any child should be less or equal to maximal rate of the parent

- $MIR(\text{parent}) \geq MIR(\text{child1})$
- $MIR(\text{parent}) \geq MIR(\text{child2})$
- $MIR(\text{parent}) \geq MIR(\text{childN})$

## HTB priority

- Works only for child queues to differentiate them.
- 8 is the lowest priority, 1 is the highest priority.
- Queue with higher priority will get the chance to satisfy its max-limit before other queues.
- Actual traffic prioritization will only work if limits are specified. A queue without limits will not prioritize anything.





# QoS example scenario HTB



## Gigaset N510 IP Pro

- 6 DECT handsets
- 6 SIP registrations
- 4 simultaneous SIP calls

# QoS example scenario HTB



# Qualify VoIP traffic with Mangle

- Identifying VoIP traffic
- Marking VoIP connections
- Marking VoIP packets within connection



Reduce CPU load

## VoIP call consists of 2 kinds of traffic

- Call setup and signalling (UDP 5060)
- RTP Voice payload (UDP ports depended on VoIP system)

# Qualify VoIP traffic with Mangle

## VoipBuster SIP settings:

- SIP port : 5060
- Registrar : sip.voipbuster.com
- Proxy server : sip.voipbuster.com

## Gigaset N510 IP Pro RTP port numbers:



The screenshot shows a configuration window titled "Listen ports voor VoIP-verbindingen". It contains a radio button selection for "Willekeurige poorten gebruiken:" with "Ja" selected. Below this are two rows of port range inputs: "SIP-poort:" with values 5060 and 5076, and "RTP-poort:" with values 5004 and 5020.

Listen ports voor VoIP-verbindingen	
Willekeurige poorten gebruiken:	<input checked="" type="radio"/> Ja <input type="radio"/> Nee
SIP-poort:	5060 - 5076
RTP-poort:	5004 - 5020

# Qualify VoIP traffic with Mangle

The screenshot shows a window titled "Firewall Address List <VoipBuster SIP>". It contains the following fields and buttons:

- Name: VoipBuster SIP
- Address: sip.voipbuster.com
- Timeout: (empty)
- Creation Time: Apr/05/2019 11:35:54
- Buttons: OK, Cancel, Apply, Disable, Comment, Copy, Remove

At the bottom left of the window, the status "enabled" is displayed.

**Add address list  
for discovery SIP  
server addresses  
Voipbuster**

	VoipBuster SIP	sip.voipbuster.com		Apr/05/2019 11:35:54
	::: sip.voipbuster.com			
D	VoipBuster SIP	77.72.169.134		Apr/05/2019 11:35:54
	::: sip.voipbuster.com			
D	VoipBuster SIP	77.72.169.129		Apr/05/2019 11:35:54

# Mark VoIP signalling connections

The image displays three overlapping Mikrotik WinBox windows for configuring Mangle Rules:

- Mangle Rule <5060> (General tab):** Chain: forward, Dst. Port: 5060.
- Mangle Rule <77.72.169.134:5060> (General tab):** Dst. Address List: VoipBuster SIP.
- Mangle Rule <5060> (Action tab):** Action: mark connection, New Connection Mark: SIP signalling conn, Passthrough: checked.

# Mark VoIP signalling packets

New Mangle Rule

General Advanced Extra Action Statistics

Chain: forward

Src. Address:

Dst. Address:

Protocol:

Src. Port:

Dst. Port:

Any. Port:

In. Interface:

Out. Interface:

In. Interface List:

Out. Interface List:

Packet Mark:

Connection Mark:  SIP signalling conn

Routing Mark:

OK

Cancel

Apply

Disable

Comment

Copy

Remove

Reset Counters

Reset All Counters

New Mangle Rule

General Advanced Extra Action Statistics

Action: mark packet

Log

Log Prefix:

New Packet Mark: SIP packet

Passthrough

OK

Cancel

Apply

Disable

Comment

Copy

Remove

Reset Counters

Reset All Counters

# Mark VoIP RTP connections

The image shows two overlapping windows from the Mikrotik WinBox interface. The background window is titled "New Mangle Rule" and has tabs for "General", "Advanced", "Extra", "Action", and "Statistics". The "General" tab is active, showing the following configuration:

- Chain: forward
- Src. Address: (empty)
- Dst. Address: (empty)
- Protocol:  udp
- Src. Port:  5004-5020
- Dst. Port: (empty)
- Any. Port: (empty)
- In. Interface: (empty)
- Out. Interface: (empty)

The foreground window is titled "Mangle Rule <5004-5020>" and also has tabs for "General", "Advanced", "Extra", "Action", and "Statistics". The "Action" tab is active, showing the following configuration:

- Action: mark connection
- Log
- Log Prefix: (empty)
- New Connection Mark: RTP conn
- Passthrough

Both windows have "OK", "Cancel", "Apply", "Disable", "Comment", "Copy", "Remove", "Reset Counters", and "Reset All Counters" buttons.



# Mark VoIP RTP packets

Mangle Rule <>

General | Advanced | Extra | Action | Statistics

Chain: forward

Src. Address: [ ]

Dst. Address: [ ]

Protocol: [ ]

Src. Port: [ ]

Dst. Port: [ ]

Any. Port: [ ]

In. Interface: [ ]

Out. Interface: [ ]

In. Interface List: [ ]

Out. Interface List: [ ]

Packet Mark: [ ]

Connection Mark:  RTP conn

Routing Mark: [ ]

OK  
Cancel  
Apply  
Disable  
Comment  
Copy  
Remove  
Reset Counters  
Reset All Counters

Mangle Rule <>

General | Advanced | Extra | Action | Statistics

Action: mark packet

Log

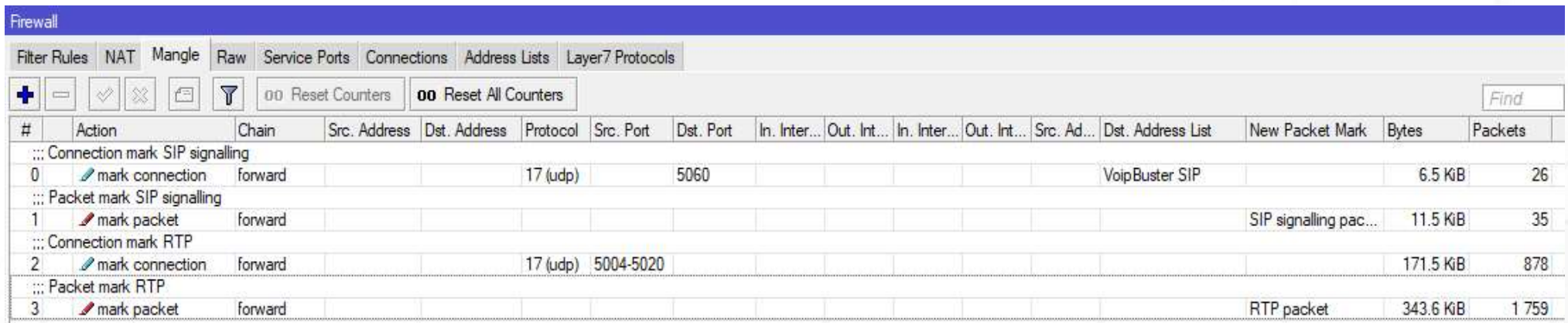
Log Prefix: [ ]

New Packet Mark: RTP packet

Passthrough

OK  
Cancel  
Apply  
Disable  
Comment  
Copy  
Remove  
Reset Counters  
Reset All Counters

# Qualify VoIP traffic with Mangle



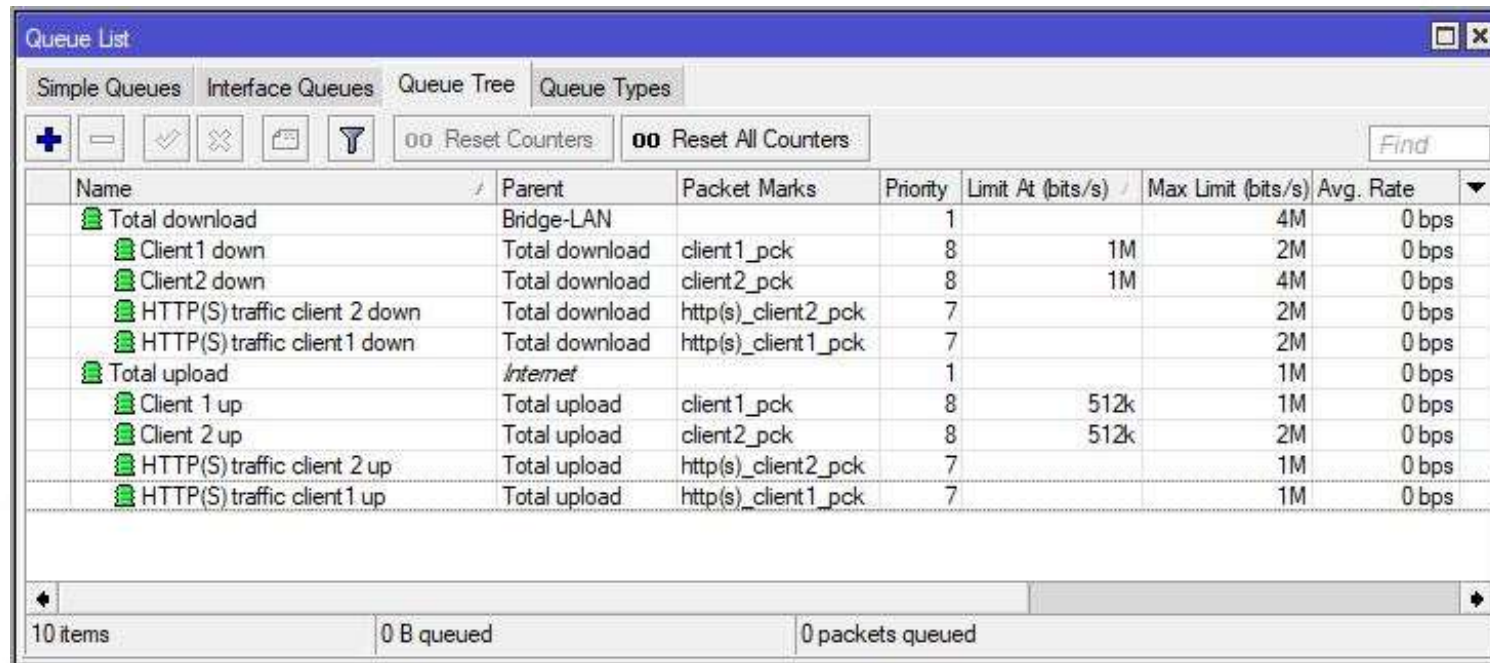
The screenshot shows the Mikrotik WinBox Firewall configuration interface, specifically the Mangle tab. The interface includes a toolbar with icons for adding, deleting, and saving rules, as well as buttons for 'Reset Counters' and 'Reset All Counters'. A search bar is located on the right side of the toolbar. Below the toolbar is a table listing the configured mangle rules. The table has columns for rule number, action, chain, source and destination addresses, protocol, source and destination ports, and statistics for bytes and packets. The rules are as follows:

#	Action	Chain	Src. Address	Dst. Address	Protocol	Src. Port	Dst. Port	In. Inter...	Out. Int...	In. Inter...	Out. Int...	Src. Ad...	Dst. Address List	New Packet Mark	Bytes	Packets
::: Connection mark SIP signalling																
0	mark connection	forward			17 (udp)		5060						VoipBuster SIP		6.5 KiB	26
::: Packet mark SIP signalling																
1	mark packet	forward												SIP signalling pac...	11.5 KiB	35
::: Connection mark RTP																
2	mark connection	forward			17 (udp)	5004-5020									171.5 KiB	878
::: Packet mark RTP																
3	mark packet	forward												RTP packet	343.6 KiB	1 759

**Provide the mangle rules with clear comments!**

# HTB implementation Queue Tree

- Queue Tree is a direct implementation of HTB



The screenshot shows the 'Queue List' window in Mikrotik WinBox. The 'Queue Tree' tab is selected. The table below displays the configuration for the Queue Tree, showing a hierarchical structure of queues. The 'Parent' column indicates the hierarchy, starting from 'Bridge-LAN' for downloads and 'Internet' for uploads. The 'Limit At (bits/s)' column shows the rate limit for each queue, and the 'Max Limit (bits/s)' column shows the maximum rate limit. The 'Avg. Rate' column shows the current average rate for each queue.

Name	Parent	Packet Marks	Priority	Limit At (bits/s)	Max Limit (bits/s)	Avg. Rate
Total download	Bridge-LAN		1		4M	0 bps
Client1 down	Total download	client1_pck	8	1M	2M	0 bps
Client2 down	Total download	client2_pck	8	1M	4M	0 bps
HTTP(S) traffic client 2 down	Total download	http(s)_client2_pck	7		2M	0 bps
HTTP(S) traffic client1 down	Total download	http(s)_client1_pck	7		2M	0 bps
Total upload	Internet		1		1M	0 bps
Client 1 up	Total upload	client1_pck	8	512k	1M	0 bps
Client 2 up	Total upload	client2_pck	8	512k	2M	0 bps
HTTP(S) traffic client 2 up	Total upload	http(s)_client2_pck	7		1M	0 bps
HTTP(S) traffic client1 up	Total upload	http(s)_client1_pck	7		1M	0 bps

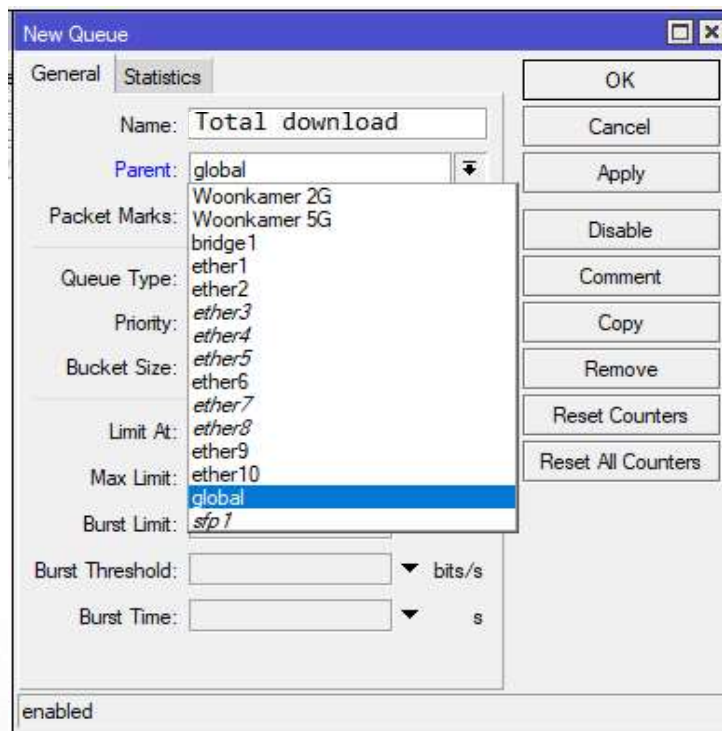
10 items      0 B queued      0 packets queued

## HTB implementation Queue Tree

- Queue Tree is one directional only and can be placed in any of the available HTBs
- Queue Tree queues don't have any order – all traffic is processed simultaneously
- All child queues must have packet marks from “/ip firewall mangle” facility assigned to them
- If placed in the same HTB, Simple queue will take all the traffic away from the Queue Tree queue

# HTB implementation Queue Tree

The Queue Tree can be placed in 'global' or in 'interface queue'



Each Interface HTB only receives traffic that will be leaving through a particular interface – there is no need for to separate upload and download in mangle

# Queue Tree downstream

The image displays four overlapping dialog boxes in Mikrotik WinBox, illustrating a queue tree configuration for downstream traffic. Each dialog box represents a different queue, with its settings as follows:

- Queue <Total downstream>**
  - Name: Total downstream
  - Parent: LAN Bridge
  - Packet Marks: (empty)
  - Queue Type: default-small
  - Priority: 8
  - Bucket Size: 0.100
  - Limit At: (empty) bits/s
  - Max Limit: 50M bits/s
  - Burst Limit: (empty) bits/s
  - Burst Threshold: (empty) bits/s
  - Burst Time: (empty) s
  - enabled
- Queue <SIP signalling down>**
  - Name: SIP signalling down
  - Parent: Total downstream
  - Packet Marks: SIP signalling packet
  - Queue Type: default-small
  - Priority: 8
  - Bucket Size: 0.100
  - Limit At: 10k bits/s
  - Max Limit: 10k bits/s
  - Burst Limit: (empty) bits/s
  - Burst Threshold: (empty) bits/s
  - Burst Time: (empty) s
  - enabled
- Queue <RTP down>**
  - Name: RTP down
  - Parent: Total downstream
  - Packet Marks: RTP packet
  - Queue Type: default-small
  - Priority: 7
  - Bucket Size: 0.100
  - Limit At: 400k bits/s
  - Max Limit: 400k bits/s
  - Burst Limit: (empty) bits/s
  - Burst Threshold: (empty) bits/s
  - Burst Time: (empty) s
  - enabled
- Queue <Other traffic down>**
  - Name: Other traffic down
  - Parent: Total downstream
  - Packet Marks: no-mark
  - Queue Type: default-small
  - Priority: 8
  - Bucket Size: 0.100
  - Limit At: (empty) bits/s
  - Max Limit: (empty) bits/s
  - Burst Limit: (empty) bits/s
  - Burst Threshold: (empty) bits/s
  - Burst Time: (empty) s
  - enabled

Each dialog box includes a 'General' tab and a 'Statistics' tab. The 'Other traffic down' dialog box also features a control panel on the right with buttons for OK, Cancel, Apply, Disable, Comment, Copy, Remove, Reset Counters, and Reset All Counters.

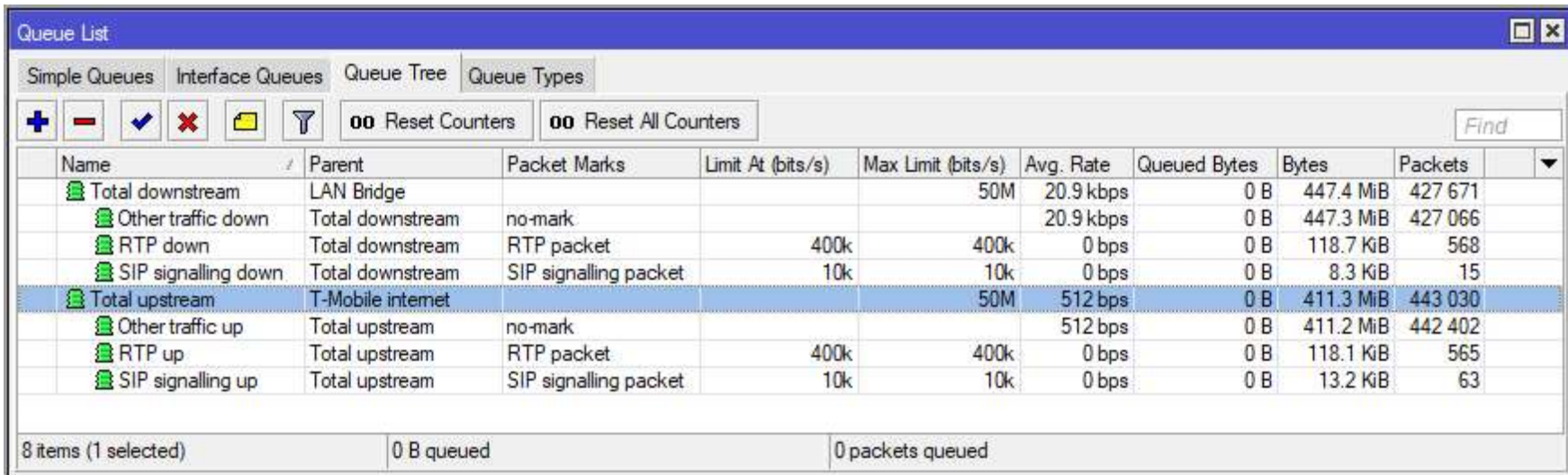
# Queue Tree upstream

The image displays four overlapping configuration windows for a Queue Tree upstream, showing a hierarchical structure:

- Queue <Total upstream>**:
  - Name: Total upstream
  - Parent: T-Mobile internet
  - Packet Marks: (empty)
  - Queue Type: default-small
  - Priority: 8
  - Bucket Size: 0.100
  - Limit At: (empty) bits/s
  - Max Limit: 50M bits/s
  - Burst Limit: (empty) bits/s
  - Burst Threshold: (empty) bits/s
  - Burst Time: (empty) s
  - enabled
- Queue <SIP signalling up>**:
  - Name: SIP signalling up
  - Parent: Total upstream
  - Packet Marks: SIP signalling packet
  - Queue Type: default-small
  - Priority: 8
  - Bucket Size: 0.100
  - Limit At: 10k bits/s
  - Max Limit: 10k bits/s
  - Burst Limit: (empty) bits/s
  - Burst Threshold: (empty) bits/s
  - Burst Time: (empty) s
  - enabled
- Queue <RTP up>**:
  - Name: RTP up
  - Parent: Total upstream
  - Packet Marks: RTP packet
  - Queue Type: default-small
  - Priority: 7
  - Bucket Size: 0.100
  - Limit At: 400k bits/s
  - Max Limit: 400k bits/s
  - Burst Limit: (empty) bits/s
  - Burst Threshold: (empty) bits/s
  - Burst Time: (empty) s
  - enabled
- Queue <Other traffic up>**:
  - Name: Other traffic up
  - Parent: Total upstream
  - Packet Marks: no-mark
  - Queue Type: default-small
  - Priority: 8
  - Bucket Size: 0.100
  - Limit At: (empty) bits/s
  - Max Limit: (empty) bits/s
  - Burst Limit: (empty) bits/s
  - Burst Threshold: (empty) bits/s
  - Burst Time: (empty) s
  - enabled
  - Buttons: OK, Cancel, Apply, Disable, Comment, Copy, Remove, Reset Counters, Reset All Counters

# HTB implementation Queue Tree

## Queue Tree structure



The screenshot shows the 'Queue List' window in Mikrotik WinBox. The 'Queue Tree' tab is selected. The table below represents the data shown in the window:

Name	Parent	Packet Marks	Limit At (bits/s)	Max Limit (bits/s)	Avg. Rate	Queued Bytes	Bytes	Packets
Total downstream	LAN Bridge			50M	20.9 kbps	0 B	447.4 MiB	427 671
Other traffic down	Total downstream	no-mark			20.9 kbps	0 B	447.3 MiB	427 066
RTP down	Total downstream	RTP packet	400k	400k	0 bps	0 B	118.7 KiB	568
SIP signalling down	Total downstream	SIP signalling packet	10k	10k	0 bps	0 B	8.3 KiB	15
Total upstream	T-Mobile internet			50M	512 bps	0 B	411.3 MiB	443 030
Other traffic up	Total upstream	no-mark			512 bps	0 B	411.2 MiB	442 402
RTP up	Total upstream	RTP packet	400k	400k	0 bps	0 B	118.1 KiB	565
SIP signalling up	Total upstream	SIP signalling packet	10k	10k	0 bps	0 B	13.2 KiB	63

8 items (1 selected) | 0 B queued | 0 packets queued

### Queue colors in Winbox:

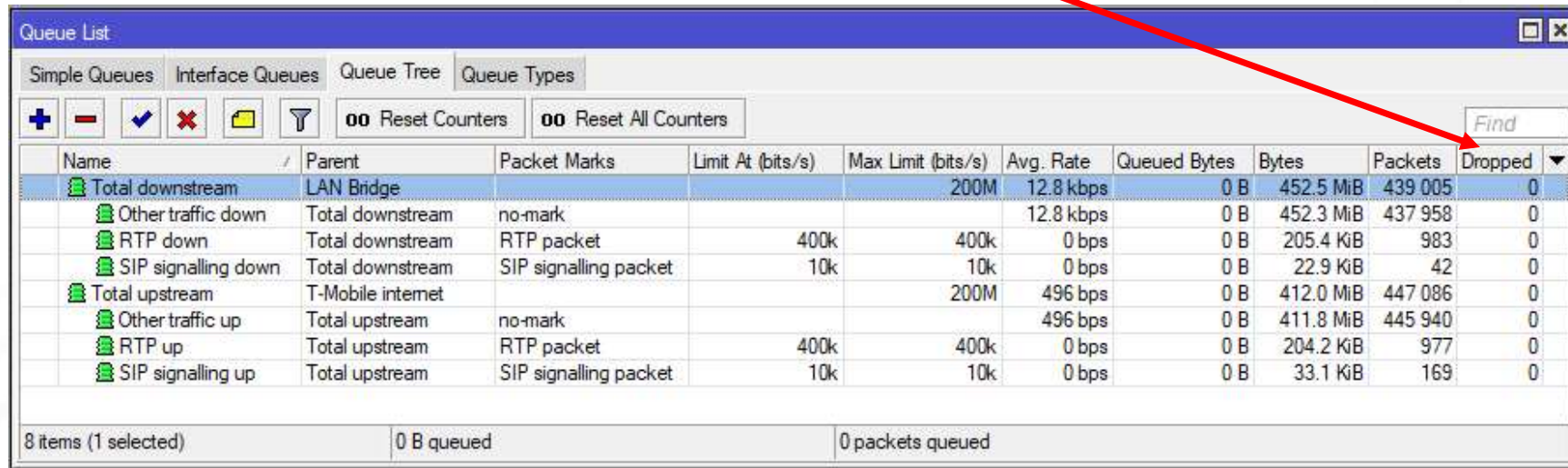
- 0% - 50% available traffic used - green
- 51% - 75% available traffic used - yellow
- 76% - 100% available traffic used - red



# Check VoIP traffic

- Checking for dropped packets in Queue Tree

Show Columns 'dropped'



Name	Parent	Packet Marks	Limit At (bits/s)	Max Limit (bits/s)	Avg. Rate	Queued Bytes	Bytes	Packets	Dropped
Total downstream	LAN Bridge			200M	12.8 kbps	0 B	452.5 MiB	439 005	0
Other traffic down	Total downstream	no-mark			12.8 kbps	0 B	452.3 MiB	437 958	0
RTP down	Total downstream	RTP packet	400k	400k	0 bps	0 B	205.4 KiB	983	0
SIP signalling down	Total downstream	SIP signalling packet	10k	10k	0 bps	0 B	22.9 KiB	42	0
Total upstream	T-Mobile internet			200M	496 bps	0 B	412.0 MiB	447 086	0
Other traffic up	Total upstream	no-mark			496 bps	0 B	411.8 MiB	445 940	0
RTP up	Total upstream	RTP packet	400k	400k	0 bps	0 B	204.2 KiB	977	0
SIP signalling up	Total upstream	SIP signalling packet	10k	10k	0 bps	0 B	33.1 KiB	169	0

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

# Check VoIP traffic by sniffing

- Activate Packet Sniffer for VoIP traffic

Packet Sniffer Settings

General Streaming Filter

Memory Limit: 100 kb

Only Headers

Memory Scroll

File Name: voipsniff

File Limit: 1000 kb

OK Cancel Apply Start Stop Packets Connections Hosts Protocols

stopped

File List

File Name	Type	Size	Creation Time
flash	disk		Jan/01/1970 02:00:04
flash/Avinco Edge Router-20181204-1446.backup	backup	76.4 KiB	Dec/04/2018 15:46:28
flash/Avinco Edge Router-20181207-1245.backup	backup	84.9 KiB	Dec/07/2018 13:45:14
flash/skins	directory		Jan/01/1970 02:00:04
flash/user-manager	directory		Nov/07/2018 03:00:02
flash/user-manager/logsqldb	file	6.0 KiB	Dec/04/2018 15:48:26
flash/user-manager/sqldb	file	80.0 KiB	Dec/04/2018 15:48:28
voipsniff	file	240.1 KiB	Apr/05/2019 15:31:26

8 items 13.8 MiB of 16.0 MiB used 14% free

# Check VoIP traffic by sniffing

The image shows two windows from a network analysis tool. The top window, titled 'voipsniff', displays a list of captured packets. The bottom window, titled 'Wireshark · RTP Streams · voipsniff', shows a detailed view of two RTP streams.

**voipsniff Packet List:**

No.	Time	Source
119	6.576084	62.41.83.78
120	6.580494	192.168.1.113
121	6.596436	62.41.83.78
122	6.600502	192.168.1.113
123	6.616406	62.41.83.78
124	6.620497	192.168.1.113
125	6.636449	62.41.83.78
126	6.640494	192.168.1.113
127	6.656436	62.41.83.78
128	6.660497	192.168.1.113
129	6.676267	62.41.83.78
130	6.680492	192.168.1.113

**Wireshark · RTP Streams · voipsniff Table:**

Source Address	Source Port	Destination Address	Destination Port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter	Mean Jitter	Status
62.41.83.78	32940	192.168.1.113	5004	0x0	g711A	415	0 (0.0%)	22.266	0.534	0.267	
192.168.1.113	5004	62.41.83.78	32940	0xcbbec9fe	g711A	412	0 (0.0%)	20.014	0.005	0.003	

# Check VoIP traffic by sniffing

Wireshark · RTP Stream Analysis · voipsniff

62.41.83.78:32940 ↔ 192.168.1.113:5004

**Forward**

- SSRC: 0x00000000
- Max Delta: 22.27 ms @ 101
- Max Jitter: 0.53 ms
- Mean Jitter: 0.27 ms
- Max Skew: -2.60 ms
- RTP Packets: 415
- Expected: 415
- Lost: 0 (0.00 %)
- Seq Errs: 0
- Start at: 6.336141 s @ 95
- Duration: 8.28 s
- Clock Drift: -222 ms
- Freq Drift: 7786 Hz (-2.68 %)

**Reverse**

- SSRC: 0xcbbec9fe
- Max Delta: 20.01 ms @ 227
- Max Jitter: 0.00 ms
- Mean Jitter: 0.00 ms
- Max Skew: 0.06 ms
- RTP Packets: 412
- Expected: 412
- Lost: 0 (0.00 %)
- Seq Errs: 0
- Start at: 6.320512 s @ 94
- Duration: 8.22 s
- Clock Drift: -223 ms
- Freq Drift: 7783 Hz (-2.72 %)

Forward to reverse start diff -0.015629 s @ -1

2 streams found. G: Go to packet, N: Next problem packet

Packet	Sequence	Delta (ms)	Jitter (ms)	Skew	Bandwidth	Marker	Status
95	5	0.00	0.00	0.00	1.60		✓
97	6	20.41	0.03	-0.41	3.20		✓
99	7	19.66	0.05	-0.07	4.80		✓
101	8	22.27	0.18	-2.34	6.40		✓
103	9	17.63	0.32	0.04	8.00		✓
105	10	20.32	0.32	-0.28			
107	11	19.86	0.31	-0.14			
109	12	20.24	0.31	-0.38			
111	13	20.54	0.32	-0.93			
113	14	21.67	0.40	-2.60			
115	15	17.53	0.53	-0.12			
117	16	20.40	0.53	-0.53			
119	17	19.42	0.53	0.06			
121	18	20.35	0.52	-0.30			
123	19	19.97	0.49	-0.27			
125	20	20.04	0.46	-0.31			
127	21	19.99	0.43	-0.30			
129	22	19.83	0.42	-0.13			
131	23	20.10	0.40	-0.22			
133	24	19.97	0.37	-0.19			
135	25	19.82	0.36	-0.01			
137	26	20.11	0.34	-0.12			
139	27	19.96	0.33	-0.08			
141	28	20.18	0.32	-0.26			
143	29	20.10	0.30	-0.36			
145	30	19.96	0.29	-0.32			
147	31	19.95	0.27	-0.27			
149	32	20.83	0.31	-1.10			

Wireshark · RTP Player

Source Address	Source Port	Destination Address	Destination Port	SSRC	Setup Frame	Packets	Time Span (s)	Sample Rate (Hz)	Payloads
62.41.83.78	32940	192.168.1.113	5004	0x00000000	62	415	6.34 - 14.6 (8.28)	8000	g711A
192.168.1.113	5004	62.41.83.78	32940	0xcbbec9fe	62	412	6.32 - 14.5 (8.22)	8000	g711A

Output Device: Default Output Device

Jitter Buffer: 50 Playback Timing: Jitter Buffer  Time of Day

Close Help

**Thank you for your attention!**