https://wiki.teltonika-networks.com/view/RUT240_Traffic_Shaping

RUT240 Traffic Shaping

The information in this page is updated in accordance with firmware version **RUT2_R_00.07.06.8**.

***Note**: **click here** for the old style WebUI (FW version RUT2XX_R_00.01.14.7 and earlier) user manual page.*

□

Contents

- 1 Summary
- 2 SQM
  - 2.1 General Setup
  - 2.2 Advanced Settings
  - 2.3 Additional notes
- 3 QoS
  - 3.1 Interfaces
  - 3.2 Classification Rules

Summary

Traffic Shaping is a bandwidth management technique used on computer networks which delays some or all datagrams to bring them into compliance with a desired traffic profile. Traffic Shaping is used to optimize or guarantee performance, improve latency, or increase usable bandwidth for some kinds of packets by delaying other kinds.

This manual page provides an overview of the Traffic Shaping pages in RUT240 devices.

If you're having trouble finding this page or some of the parameters described here on your device's WebUI, you should **turn on "Advanced WebUI" mode**. You can do that by clicking the "Advanced" button, located at the top of the WebUI.

SQM

The **SQM** page is used to set up Smart Queue Management (SQM) instances which can limit the download and upload speeds of selected network interfaces.

Note: SQM is additional software that can be installed from the **System → Package Manager** page.

The **Smart Queue Management** section contains a list of existing SQM instances and can be used manage them (edit/delete/enable/disable). By default the list is empty:



To add a new instance and begin editing, look below to the **Add New Instance** section and:

1. enter a custom name in the 'Name' field;
2. click the 'Add' button;
3. configuration window will open automatically.



Refer to the sections below for information on editing SQM instances.

General Setup

The **General Setup** section is used to set download and upload speed limits for a selected network interface.



Field	Value	Description
Enabled	off | on; default: **off**	Turns the SQM configuration on or off.
Interface name	network interface; default: **br-lan (lan)**	Selects to which interface this SQM configuration will apply. **Additional note:** consider reading the Additional notes section before limiting speeds on selected interfaces.
Download speed (kbit/s)	integer [0..2147483647]; default: **none**	Limits the download speed (ingress) of the selected interface to the value specified in this field.
Upload speed (kbit/s)	integer [0..2147483647]; default: **none**	Limits the upload speed (egress) of the selected interface to the value specified in this field.

Advanced Settings

The **Advanced Settings** section describes the internal logic of the SQM configuration.



Field	Value	Description
Queuing disciplines usable on this system	cake | fq_codel; default: **cake**	Queue management logic type. Look to the row below for detailed information.

Queue setup script	layer_cake.qos | piece_of_cake.qos | simple.qos | simplest.qos | simplest_tbf.qos; default: **piece_of_cake.qos**	Describes the queue management logic. • **cake** - **layer_cake.qos** - uses the cake qdisc as a replacement for both htb as shaper and fq_codel as leaf qdisc. This exercises cake's diffserv profile(s) as different "layers" of priority. - **piece_of_cake.qos** - uses the cake qdisc as a replacement for both htb as shaper and fq_codel as leaf qdisc. It just does not come any simpler than this, in other words it truly is a "piece of cake". • **fq_codel** - **simple.qos** - BW-limited three-tier prioritisation scheme with your qdisc on each queue. - **simplest.qos** - simplest possible configuration: HTB rate limiter with your qdisc attached. - **simplest_tbf.qos** - simplest possible configuration (TBF): TBF rate limiter with your qdisc attached. TBF may give better performance than HTB on some architectures.

Additional notes

Please consider that limiting the speed of certain interfaces may limit the speed of other interfaces as well. For example, if you limit the download and upload speeds of the default LAN interface (*br-lan*) to 50 000 kbps (50 Mbps) this will affect the speeds between:

- Ethernet LAN machines (including the device)
- Ethernet LAN machines and WAN*
- WiFi devices**
- Wifi devices and LAN machines**
- WiFi devices and WAN**

* Since LAN machines communicate with WAN servers over the device which is also considered a LAN machine, LAN-to-WAN and WAN-to-LAN speeds are also affected.

** By default, wireless network interfaces are attached to the default Ethernet LAN network (*br-lan*). If you wish for WiFi speeds to remain unaffected, consider creating an additional Ethernet interface and attach your WiFi interfaces to it. This can done in the following WebUI pages:

- Network → Interfaces (to create an additional network interface)
- Network → Wireless (to attach WiFi interfaces to Ethernet interfaces)

The table below demonstrates speed changes between some of the main default network interfaces if ***br-lan* speed is set to 50 Mbps**.

	Ethernet LAN	**WiFi LAN**	**Ethernet WAN**	**Mobile WAN**
Ethernet LAN	50	50	50	50
WiFi LAN	50	50	50	50
Ethernet WAN	50	50	unchanged	unchanged

Mobile WAN	50	50	unchanged	unchanged

Changing the speeds of other network interfaces may also affect more than one interface. However, the Ethernet LAN interface (*br-lan*) is related to most interfaces. Therefore, it is advised to limit the speeds of other interfaces in order to achieve more precise results (unless, of course, your goal to is to limit the speed of *br-lan*).

Another example to consider is limiting WAN interface speeds since that is the most widely implemented use case and affects less interface-to-interface relationships. For instance, limiting **Ethernet WAN (*eth1*) speed to 50 Mbps** would produce the following results:

	Ethernet LAN	**WiFi LAN**	**Ethernet WAN**	**Mobile WAN**
Ethernet LAN	unchanged	unchanged	50	unchanged
WiFi LAN	unchanged	unchanged	50	unchanged
Ethernet WAN	50	50	unchanged	unchanged
Mobile WAN	unchanged	unchanged	unchanged	unchanged

Please take caution when using SQM to limit speeds in order to avoid unexpected results.

QoS

QoS provides the possibility to prioritize network traffic based on hosts, ports or services and limit download & upload speeds on a selected interface.

This chapter of the user manual provides an overview of the QoS page in RUT240 devices.

Note: QoS is additional software that can be installed from the **System → Package Manager** page.

Interfaces

The **Interfaces** section is used to set up download and upload speed limits on the device's network interfaces.



Field	Value	Description
Calculate overhead	yes | no; default: **no**	Decreases upload and download ratio to prevent link saturation.
Download speed (kbit/s)	integer; default: **none**	Maximum download speed for the specified interface.
Upload speed (kbit/s)	integer; default: **none**	Maximum upload speed for the specified interface.
Enable	yes | no; default: **no**	Turns the rule on or off.

Classification Rules

The **Classification Rules** section is used to configure rules that prioritize certain traffic over other, less important traffic. This traffic can be defined by a source or destination host, port or a network protocol. Traffic that matches a rule with higher preference will be handled with higher priority.

Generally, this should be used in cases where you want to isolate certain types of traffic in order to

make sure the RUT240 is handling more important traffic (for example, video streaming, SSH) at a higher priority and less important traffic (FTP, HTTP) at a lower priority.



Field	Value	Description
Target	Priority | Express | Normal | Bulk; default: **Normal**	Defines the priority of handling the type of traffic defined in this rule.
Source host	All | IP | -- Custom --; default: **All**	Host(s) from which data will be transmitted.
Destination host	All | IP | -- Custom --; default: **All**	Host(s) to which data will be transmitted.
Protocol	All | TCP | UDP | ICMP | -- Custom --; default: **All**	Data transmission protocol to match the rule.
Ports	All | integer [1..65535]; default: **none**	Port number(s) to match the rule. You can enter multiple ports by separating them by commas (e.g., *22,53,80*).