

You are here: [DD-WRT wiki mainpage](#) / [Web-GUI](#) / [NAT/QoS](#) / [QoS](#)

[English](#) • [Deutsch](#) • [Español](#) • [Français](#) • [Italiano](#) • [???](#) • [Polski](#) • [Português](#) • [???????](#) • [Svenska](#) • [???\(????\)?](#) • [???\(??\)?](#) •

## Contents

- [1 Introduction](#)
- [2 Priorities explained](#)
  - ◆ [2.1 TCP Packet Priority](#)
- [3 Precedence](#)
- [4 Initial Setup](#)
- [5 Prioritizing by Application \(Skype, Http\) or Port Range \(P2P\)](#)
- [6 Prioritizing by Interface](#)
- [7 Prioritizing by Netmask \(IP address\)](#)
- [8 Prioritizing by MAC Address](#)
- [9 How Do You Check What QoS Priorities Were Applied](#)
- [10 Time Based QoS](#)
- [11 External Links](#)

## Introduction

**Be using a build NO OLDER than r32170 before proceeding!**

Quality of Service (**QoS**) is a method to guarantee a bandwidth relationship between individual applications or protocols. This is very handy when you max out your connection so that you can allow for each application to have some bandwidth and so that no single application can take down the internet connection. This allows, for example, a full speed download via FTP without causing jittering on a VOIP chat. The FTP will slow down slightly as bandwidth is needed for the VOIP, provided VOIP was given greater priority.

Please note, as of 336XX, if QoS is enabled, SFE (Shortcut Forwarding Engine) is disabled, even if it shows up as enabled in the GUI, it is disabled.

**If you plan on using QoS, please read [Priorities explained](#) and [Precedence](#) before going any farther.**

## Priorities explained

- **Maximum** - This class offers maximum priority and should be used sparingly.
- **Premium** - Second highest bandwidth class, by default handshaking and ICMP packets fall into this class. Most VoIP and video services will function good in this class if Express is insufficient.
- **Express** - The Express class is for interactive applications that require bandwidth above standard services so that interactive apps run smoothly.
- **Standard** - All services that are not specifically classed will fall under standard class.
- **Bulk** - The bulk class is only allocated remaining bandwidth when the remaining classes are idle. If the line is full of traffic from other classes, Bulk will only be allocated 1% of total set limit. Use this

## Quality\_of\_Service

class for P2P and downloading services like FTP.

Bandwidth is allocated based on the following "minimum to maximum" percentages of downlink and uplink values for each class as of current builds:

- Maximum: 75% - 100%
- Premium: 50% - 100%
- Express: 25% - 100%
- Standard: 15% - 100%
- Bulk: 5% - 100%

What this means is that if you have 10,000kbit of uplink traffic, "Standard" class traffic can be reduced and de-prioritized to 15% or 1,500kbit when a concurrent express or higher priority service requires the down/uplink pipe at the same time.

Check which priorities are used with the command below:

```
iptables -t mangle -vnL
```

Then scroll down to the Chain SVQOS\_SVCS section.

## TCP Packet Priority

Builds before r21061 will not have this option. Update your build if you dont have it, stay up to date.

Prioritize small TCP-packets with the following flags: ACK/SYN/FIN/RST

For detailed info on what these packets do see:

[http://en.wikipedia.orghttp://wiki.dd-wrt.com/wiki/Transmission\\_Control\\_Protocol#TCP\\_segment\\_structure](http://en.wikipedia.orghttp://wiki.dd-wrt.com/wiki/Transmission_Control_Protocol#TCP_segment_structure)

It is highly recommended to have at least, SYN, FIN & RST checked, OR, none at all. ACK can go both ways as P2P intensive applications such as uTorrent etc involve a lot of ACKs, so theoretically prioritizing ACKs means you "prioritized P2P" though that is not entirely' accurate. Read up & do your own testing to find out whats best for your network. If you do not do large amounts of P2P activity on your network or none at all, then enable ACK prioritization.

## Precedence

With all these ways of marking traffic its easy to get confused about how seemingly contradictory requirements are resolved. For example, what happens if you have an IP rule setting IP 192.168.1.2 to priority "maximum" and have a MAC rule setting AA:BB:CC:DD:EE:FF to priority "bulk"?

## Quality\_of\_Service

The order the precedence is as follows:

- (1st) MAC Priority
- (2nd) Netmask Priority
- (3rd) Interface Priority
- (4th) Services Priority
- (5th) Ethernet Port Priority

**NOTE:** Ethernet Port Priority only works on old 802.11g only models with ADMtek switch chips. If you don't have ethernet port priority listed, your router does not support it. Ethernet port priority is different than interface priority.

**NOTE:** Services can be used at the same time as netmask or MAC, such as limiting 192.168.1.2 to 6 Mbps down & 512 Kbps up while having http set to express, that device will have http packets prioritized within it's allocated bandwidth limit. This only applies to builds r21061 & newer

For netmask, the IP address entries are applied in the order that they appear in your netmask table. Only the first match applies. Example, if you have an entry marking 192.168.1.10/32 as bulk, followed by an entry ABOVE IT, marking 192.168.1.0/24 (all 192.168.1.X) as premium, the traffic from 192.168.1.10 would be marked bulk because it was the first match.

For services, The services entries are applied in the order that they appear in your services tables going from bottom to top. Again, only the first match will apply.

## Initial Setup

- Log into the Web Interface
- Select the NAT/QoS tab and then the QoS sub-tab.
- Click "Enable"
- Set Port to "WAN". This works for all QoS setups EXCEPT, when using QoS by interface on a BRIDGED interface under "interface priority", UNBRIDGED interfaces work fine with WAN port setting. If you want to use QoS on a BRIDGED interface you must select port as "LAN & WLAN", which also works for all other QoS setups, but with slightly more CPU usage.
- Select HTB as your packet scheduler if you have "queuing discipline" listed below it, if not then use HFSC.
- Select FQ\_CODEL as your queuing discipline.
- Set your download and upload speeds. You can use a speed test like Speedtest.net or dslreports.com/speedtest to check your actual connection speed. Some ISPs also provide their own bandwidth testing service, which may be more reliable than the links provided. Enter no higher than 95% of the values you measured into the proper fields. After you have everything set run the speed test again. If you get near 90% of your previous measurement in each direction then things are cool. If

## Quality\_of\_Service

you get results which are way off then chances are that you have reversed these values. You must enter a value for the uplink field but if you want you can enter 0 for the downlink field in which case no QoS will occur in that direction, setting your downlink field to 0 isn't recommended.

It probably bugs you to set less than 100% of your available bandwidth in these fields but this is required. There will be a bottleneck somewhere in the system and QoS can only work if the bottleneck is in your router where it has control. The goal is to force the bottleneck to be in your router as opposed to some random location out on the wire over which you have no control. Some ISP's even have bursting ("powerboost") which will temporarily give you extra bandwidth when you first start using your connection but will later throttle down to a sustained rate. Fortunately there is usually a minimum level that you receive on a consistent basis and you must set your QoS limits below this minimum. The problem is finding this minimum and you may have to repeat speed tests many times before determining it. For this reason start with 80% of your measured speed and try things for a couple of days. If the performance is acceptable you can start to inch your levels up. If you go even 2% higher than you should be, your QoS will totally stop working (just too high) or randomly stop working (when your ISP node/DSLAM is slow aka saturated). This can lead to a lot of confusion on your part so get it working first by conservatively setting these speeds and then optimize later.

## Prioritizing by Application (Skype, Http) or Port Range (P2P)

- Choose an available Service or Port Range from the list or create one, and then press "Add" next to it.
- For P2P Applications, due to evolving protocols, encryption and obfuscation, it can be much better to define a port range [such as TCP/UDP, 60000-61000]. Set your P2P applications to operate within this range. This can significantly reduce the load on the router, avoid mis-identifying packets, and more efficiently shape your network traffic.
- Add all your other selected Services and Port Ranges here
- Choosing a Layer7 service based entry can work better than choosing a port range; though the router works harder as it has to dig into the packets beyond the header, to look at the data they contain.

If you wish to add more than one priority then use the "Add" button to create more entries.

## Prioritizing by Interface

Select your preferred interference, click add, then select the speed or priority you want. You can also limit ethernet ports this way as well (ethX or vlanX). Any limits or priorities set are shared for that interface regardless how many clients are connected to it. Excellent for running a guest network/hotspot on eg, ath1.1, applying QoS on the entire interface makes it impossible for a greedy user to bypass it by MAC cloning, changing IPs etc, short of connecting to a different interface. The same interface can also be entered multiple times with different speed limits or priorities for different services, example, ath0 512/512 with ssl & ath0 0/1024 with http would mean ssl traffic on ath0 is limited to 512kbps down & up, http is unlimited on down (up to global limits is used) & limited to 1024 (1mbps) on up, remaining entered services are not limited (up to global limits for both directions).

## Prioritizing by Netmask (IP address)

These are entered in CIDR notation including the network prefix.

For example, to specify a single IP address enter xxx.xxx.xxx.xxx/32. Be careful to enter netmask as /32 because leaving it /0 means ALL IPs!.

The netmask is the number of bits of the IP address to match. For example, the entry 192.168.1.0/24 matches 192.168.1.x addresses. An entry of 192.168.0.0/16 matches 192.168.x.x addresses. If you're unsure of how to create CIDR subnet masks and what they mean, then use a [subnet calculator](#).

After you have filled it out, press "add" next to it. If you want to add multiple entries (make sure to have order correct!) click "save" before entering in another so any previous changes don't get deleted, only click "apply" when you want to start testing your current changes displayed.

## Prioritizing by MAC Address

In the case you want to prioritize traffic from a particular device without a static IP address on your LAN, you can prioritize by MAC Address. Enter the MAC Address of the device and press "Add" next to it.

## How Do You Check What QoS Priorities Were Applied

The DDWRT web UI doesn't display any live traffic. Short of doing a practical test, you can get your hands dirty by checking the conntrack entries via telnet or ssh access in the router. When you're logged in run:

```
iptables -t mangle -vnL
```

Then scroll down to the Chain SVQOS\_SVCS section.

With the above iptables mangle command you can see the inbound/outbound chains, entered IPs/MACs/services & whats being matched where.

It will list out all currently open connection and protocol that is currently being routed by the router. This is what it would look like:

```
tcp      6 113 ESTABLISHED src=192.168.1.5 dst=83.141.4.210 sport=48959 dport=21 src=83.141.4.210
udp      17 29 src=192.168.1.5 dst=128.63.2.53 sport=56105 dport=53 src=128.63.2.53 dst=216.239.6
```

What you'll be interested to look at will be the first set of source and destination IP, including the port numbers. Next the presence of l7proto and the "mark" field. The entries indicate the current live connection QoS priority applied on them based on the "mark" field. The "mark" values corresponds to the following:

## Quality\_of\_Service

- Maximum: 100
- Premium: 10
- Express: 20
- Standard: 30
- Bulk: 40
- (no QoS matched): 0

You may see "mark=0" for some l7proto service even though they are configured in the list of QoS rules. This may mean that the layer 7 pattern matching system didn't match a new or changed header for that protocol. Custom service on port matches will usually take care of these.

## Time Based QoS

As described in [this thread](#) you can use [CRON](#) jobs to enable/disable QoS. This is just a simplistic approach but more complex things could be done if you put your mind to it. These commands will enable HTB QoS on the WAN port from 5PM to 1AM but you will still need to configure everything else in the GUI. If you want to use LAN&WLAN then change "`get_wanface``" to "`imq1`". To change the times, see the [CRON](#) page for information.

```
1 17 * * * root /usr/sbin/svqos `nvram get wshaper_downlink` `nvram get wshaper_uplink` `get_wanface`
*/6 1-17 * * * root /usr/sbin/svqos stop 0 `get_wanface` 0 0
```

If you use HFSC then you would do something like this instead.

```
1 17 * * * root /usr/sbin/svqos2 `nvram get wshaper_uplink` `nvram get wshaper_downlink` `get_wanface`
*/6 1-17 * * * root /usr/sbin/svqos2 stop 0 `get_wanface` 0 0
```

As described in [this thread](#) you can also set different rates at different times by doing something like this which changes the HTB rates.

```
1 23 * * * root /usr/sbin/svqos [downlink rate] [uplink rate] `get_wanface` `nvram get wan_mtu` 0
1 10 * * * root /usr/sbin/svqos [downlink rate] [uplink rate] `get_wanface` `nvram get wan_mtu` 0
```

**Edit: This will cause trouble on current firmware releases greater than r21061. Use the predefined service handler to stop/start QoS instead. Even if needed, use imq1 instead of br0 for internal traffic shaping.**

```
stopservice wshaper
startservice wshaper
```

**If you need to alter down/up rates edit the nvram variables before restarting wshaper**

```
nvram set wshaper_downlink=20000
nvram set wshaper_uplink=800
startservice wshaper
```

## External Links

- [HTB Manual](#)
- [iptables man page](#)
- [L7-Filter](#)
- [Frottle sourceforge](#)
- [Lots of QoS myths explained, what QoS can & can't do & much more info.](#)