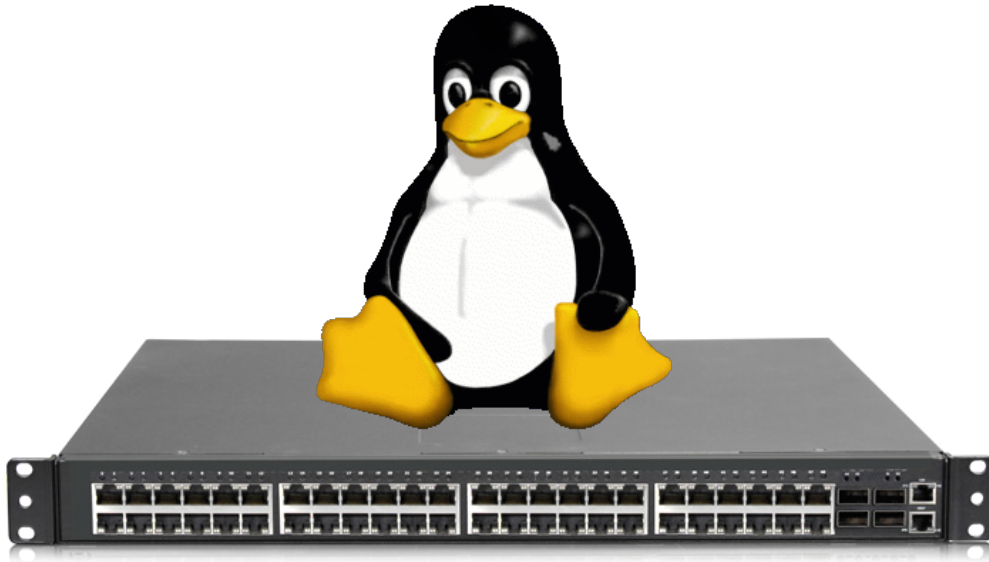


Why and How to Increase and decrease txqueuelen (Transmit Queue Length) in GNU / Linux

Author : admin



In **GNU / Linux network routers** and sometimes even home PCs it is mostly helpful to play with a **TCP / IP stack Network Interface Card (NIC)** value called (**txqueuelen**) in order to make the Network interface (Ethernet device) to work better with the type of network that is connected to it. On a slower LAN or Internet connections **txqueuelen** is better to be decreased whether on a high speed connections raising it will increase the performance of network traffic to perform large homogenous network transfers.

Therefore on a small private networks in many occasions (though it depends on the served services and exact type of network) it is more useful to decrease the txqueuelen Linux value whether on a High latency networks increasing it will reap a great benefits for your ISP or Hosting routers.

So what is txqueuelen value instructs the kernel on the largeness of transmit queue of the network interface device.

1. Decrease the number of txqueuelen in GNU / Linux on a slower networks (ADSL routers), Mobile networks etc to improve network latency

For a slower router devices be it WI-FI routers such as D-Link or any cheap chinese Wifi or LAN or Telephone or ISP company networks that provide internet via ADSL routers that run Linux or has access to Linux console with available access to ***ifconfig command or have an iproute2 Linux package installed (most of them have)*** it is very helpful to reduce its size for a smaller values in order to guarantee a network high latency.

Therefore reducing the txqueuelen value to a number like **200 for ADSL** provided internet can benefit you. Assuming that you know the NIC name in Linux the first one is usually eth0 in order to reduce this **value to 200** issue:

```
ifconfig eth0 txqueuelen length
ifconfig eth0 txqueuelen 200
```

To do the same with iproute2 (**ip**) command in case if the router is missing ifconfig or you just prefer to use iproute2 (advanced and newer bunch of commands that improved the Linux networking functionalities thanks to Russian Alexei Kuznecov 15 years ago or so) run:

```
ip link set eth0 txqueuelen length
ip link set eth0 txqueuelen 200
```

To make it permanent you can either create a brand new rc. script lets say **/etc/init.d/rc.queuelenght** and add the commands or use the good old **/etc/rc.local** to make the commands get loaded on every GNU / linux or router boot time.

Above reduction will make your Network TCP stack more responsive.

2. Increase the number of txqueuelen in GNU / Linux to improve the network latency on a company NAT or other ISP routers

Depending on the scale of Internet bandwidth that is coming inside your router you should decide how much you would like to increase the **txqueuelen value**.

Generally speaking txqueuelen value could be played with on a Large IN / OUT Company Routers with values between 1000 and 20000

Most GNU / Linux distributions comes preconfigured as a default a value of **1000**, so lets say you would like to raise the value to **5000** to improve network transfer latency, here is how to do it via **ifconfig**:

```
ifconfig eth0 txqueuelen length  
ifconfig eth0 txqueuelen 5000
```

With **IPRoute2** analogously run:

```
ip link set eth0 txqueuelen length  
ip link set eth0 txqueuelen 5000
```

Of course the best way to figure out what will be the best value for your case is to experiment a little bit with the value and use some kind of [speedtest \(test bandwidth service - this one is mine\)](#) from the many available online.

You have to consider the type of hardware of Server and most importantly the Network Card hardware (which vendor and what are the recommended ones from the vendor).

If the router is some Linux distribution lets say a **Debian / Ubuntu Linux etc other deb based one**, it is better to make the necessary permanent changes to make them be loaded on system boot not via `/etc/rc.local` but by adding to file `/etc/network/interfaces`:

```
vim /etc/network/interfaces
```

```
up ifconfig $IFACE txqueuelen 8000
```

Note that as network traffic processing depends on CPU power, decreasing the **txqueuelen** will add a little bit of extra load to the Central Processing Unit.

In RHEL / CentOS Linux and SuSE server a good way to increase txqueuelen network latency is by running:

```
# cat /etc/rules.d/71-net-txqueuelen.rules
SUBSYSTEM=="net", ACTION=="add", KERNEL=="eth*", ATTR{tx_queue_len}=="10000?
EOF
```

```
udevadm trigger
```

I've read on the internet that a lot of people are reporting a very good results with a **txqueuelen** setting of **2000** as this value is said to give good results and not hammering the CPU too hardly, so it might be a good idea if you're experimenting with values to find out which one suits you better is to start with this one, e.g. run:

```
ifconfig eth0 txqueuelen 2000  
echo 2000 > /proc/sys/net/core/netdev_max_backlog
```

Well of course keep in mind that sometimes increasing the txqueuelen to too large values will make you an easier target for Denial of Service Attack attempts, though most modern computer routers should be able to behave good even with the highest value like 10000.