How to limit network bandwidth on Linux

Author : Dan Nanni

Categories : Networking

Tagged as : bandwidthtc

If you often run multiple networking applications on your Linux desktop, or share bandwidth among multiple computers at home, you will want to have a better control over bandwidth usage. Otherwise, when you are downloading a big file with a downloader, your interactive SSH session may become sluggish to the point where it's unusable. Or when you sync a big folder over Dropbox, your roommate may complain that video streaming at her computer gets choppy.

In this tutorial, I am going to describe two different ways to rate limit network traffic on Linux.

Rate Limit an Application on Linux

One way to rate limit network traffic is via a command-line tool called `trickle`. The `trickle` command allows you to shape the traffic of any particular program by "pre-loading" a rate-limited socket library at run-time. A nice thing about `trickle` is that it runs purely in user-space, meaning you don't need root privilege to restrict the bandwidth usage of a program. To be compatible with `trickle`, the program must use `socket` interface with no statically linked library. `trickle` can be handy when you want to rate limit a program which does not have a built-in bandwidth control functionality.

To install `trickle` on Ubuntu, Debian and their derivatives:

```
$ sudo apt-get install trickle
```

To install `trickle` on Fedora or CentOS/RHEL (with EPEL repository):

```
$ sudo yum install trickle
```

Basic usage of `trickle` is as follows. Simply put, you prepend `trickle` (with rate) in front of the command you are trying to run.

```
$ trickle -d -u
```

This will limit the download and upload rate of to specified values (in KBytes/s).

For example, set the maximum upload bandwidth of your `scp` session to 100 KB/s:

```
$ trickle -u 100 scp backup.tgz alice@remote_host.com:
```

If you want, you can set the maximum download speed (e.g., 300 KB/s) of your Firefox browser by creating a custom launcher with the following command.

```
trickle -d 300 firefox %u
```

Finally, `trickle` can run in a daemon mode, where it can restrict the "aggregate" bandwidth usage of all running programs launched via `trickle`. To launch `trickle` as a daemon (i.e., `trickled`):

```
$ sudo trickled -d 1000
```

Once the `trickled` daemon is running in the background, you can launch other programs via `trickle`. If you launch one program with `trickle`, its maximum download rate is 1000 KB/s. If you launch another program with `trickle`, each of them will be rate limited to 500 KB/s, etc.

Rate Limit a Network Interface on Linux

Another way to control your bandwidth resource is to enforce bandwidth limit on a per-interface basis. This is useful when you are sharing your upstream Internet connection with someone else. Like anything else, Linux has a tool for you. `wondershaper` exactly does that: rate-limit a network interface.

`wondershaper` is in fact a shell script which uses `tc` to define traffic shaping and QoS for a specific network interface. Outgoing traffic is shaped by being placed in queues with different priorities, while incoming traffic is rate-limited by packet dropping.

In fact, the stated goal of `wondershaper` is much more than just adding bandwidth cap to an interface. `wondershaper` tries to maintain low latency for interactive sessions such as SSH while bulk download or upload is going on. Also, it makes sure that bulk upload (e.g., Dropbox sync) does not suffocate download, and vice versa.

To install `wondershaper` on Ubuntu, Debian and their derivatives:

```
$ sudo apt-get install wondershaper
```

To install `wondershaper` on Fedora or CentOS/RHEL (with EPEL repository):

```
$ sudo yum install wondershaper
```