

# curl-loader - High Load HW

<!-- -->

## Table of contents

1 .....	2
---------	---

## 1.

We have used the following HW to run loads of 40-60 K of HTTP 1.1 clients/users:

an old PC with dual Xeon and 2-4 GB of checked memory, where 4GB is recommended. We compiled curl-loader with optimization, tuned linux as described in our HOWTOS-FAQs and run the loader with threads using command line option -t 2 or -t 4. Please, note, that due to some unresolved bugs, most probably in libevent, the loader may crash on start. Workaround it by re-running and/or, optionally, changing the number of threads. We are recommending to run the number of threads close to the number of logical CPUs.

To run a load of 100-150 K HTTP 1.1 clients/users we would suggest to use:

a PC-box equipped with two modern dual core Intel or AMD CPUs and 8GB of checked RAM. Please, proceed with all optimizations as in HOWTOS-FAQs, mind the total bandwidth limitations of the loading PC NICs and run curl-loader with -t 4 command-line option. Y may wish to limit maximum bandwidth used by each client as specified in HOWTOS-FAQs, using a special URL-section tag. Please, also edit the file in curl-loader distribution patches/libevent-nevent.patch and correct NEVENT number from the set 61000 to the maximum number of connections/clients you wish to reach plus more 1000. Further run "make cleanall" and rebuild curl-loader with optimization as described by the HOWTOS-FAQs.

We would recommend to use IP\_SHARED\_NUM specified set of shared ip-addresses, where 64K IP-addresses is a limit of a single IP-address. Thus, to create a load of 200 000 clients, one may need 4 IP-addresses.

Many testing scenarios with 100K clients do not require strong processing power and a PC with less CPU resources could be enough. Each client requires about 30-40K of memory, whereas better to keep more spare memory for the system; therefore, compromises on memory are not really recommended.