Fast Start Instructions.

BUILDING CURL-LOADER.

Build it using general C development environment with bash, gcc compiler (3 or 4 series), make, etc on a linux machine.
Building pre-requirements are:
1. openssl binaries;
2. openssl development package with include files (on debian package libssl-dev);

Adjust Makefile variables to point to the openssl headers and libraries. If you want to specify an openssl development directory with include files (e.g. crypto.h), export environment variable OPENSSLDIR with the value of that directory.
For example: $export OPENSSLDIR=the-full-path-to-the-directory
Run the following commands from your bash linux shell:

```
$tar zxfv curl-loader-<version>.tar.gz
$cd curl-loader-<version>
$make
```

By default, we are building both libcurl and curl-loader without optimization and with debugging -g option. To build with optimization and without debugging, please, run:

```
$make cleanall
$make optimize=1 debug=0
```

Optionally, you can run as a root:

```
#make install
```

and enjoy our man pages by $man curl-loader and $man curl-loader-config.
A known building issue is libidn.so, which means, that some linux distributions do have some libidn.so.11 or libidn.so.10, but not libidn.so, and it breaks linking.
Resolve the issue by creating a softlink:

```
#cd to the directory, where e.g. libidn.so.11 is installed and run
#ln -s libidn.so.11 libidn.so
```

If the above was not helpful, try editing the Makefile and removing -lidn from the linking string.
If still any building issues, please, fill you free to contact us for assistance.

LOADING CONFIGURATION.

To run the load create a configuration file to be passed to curl-loader by the -f commmand line option, e.g.

```
#curl-loader -f ./conf-examples/bulk.conf
```

For more examples, please, look at the files in "conf-examples" directory. You may copy an example file and edit it.

ENVIRONMENT AND SYSTEM.

Running hundreds and thousands of clients, please, do not forget:
- to increase limit of descriptors (sockets) by running e.g.

```
#ulimit -n 10000;
```

- optionally, to set reuse of sockets in time-wait state: by setting

```
#echo 1 > /proc/sys/net/ipv4/tcp_tw_recycle and/or
#echo 1 > /proc/sys/net/ipv4/tcp_tw_reuse;
```

In some cases you may need to increase the system limits for open descriptors (sockets).

USAGE:

run as a root user:

```
#./curl-loader -f <configuration filename> [other options]
```

If curl-loader on start reports any problems with your configuration file, please, "don't panic" - fix the problems.
For more details, please, look into the FAQs page.