



Full credit is given to the above companies including the OS that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'luac.1'

\$ man luac.1

LUAC(1) General Commands Manual LUAC(1)

NAME

luac - Lua compiler

SYNOPSIS

luac [options] [filenames]

DESCRIPTION

luac is the Lua compiler. It translates programs written in the Lua programming language into binary files containing precompiled chunks that can be later loaded and executed.

The main advantages of precompiling chunks are: faster loading, protecting source code from accidental user changes, and off-line syntax checking. Precompiling does not imply faster execution because in Lua chunks are always compiled into bytecodes before being executed. luac simply allows those bytecodes to be saved in a file for later execution. Precompiled chunks are not necessarily smaller than the corresponding source. The main goal in precompiling is faster loading.

In the command line, you can mix text files containing Lua source and binary files containing precompiled chunks. luac produces a single

output file containing the combined bytecodes for all files given. Ex?

Executing the combined file is equivalent to executing the given files.

By default, the output file is named `luac.out`, but you can change this with the `-o` option.

Precompiled chunks are not portable across different architectures.

Moreover, the internal format of precompiled chunks is likely to change when a new version of Lua is released. Make sure you save the source files of all Lua programs that you precompile.

OPTIONS

`-l` produce a listing of the compiled bytecode for Lua's virtual machine. Listing bytecodes is useful to learn about Lua's virtual machine. If no files are given, then `luac` loads `luac.out` and lists its contents. Use `-l -l` for a full listing.

`-o file`

output to file, instead of the default `luac.out`. (You can use `'|'` for standard output, but not on platforms that open standard output in text mode.) The output file may be one of the given files because all files are loaded before the output file is written. Be careful not to overwrite precious files.

`-p` load files but do not generate any output file. Used mainly for syntax checking and for testing precompiled chunks: corrupted files will probably generate errors when loaded. If no files are given, then `luac` loads `luac.out` and tests its contents. No messages are displayed if the file loads without errors.

`-s` strip debug information before writing the output file. This saves some space in very large chunks, but if errors occur when running a stripped chunk, then the error messages may not contain the full information they usually do. In particular, line numbers and names of local variables are lost.

`-v` show version information.

`--` stop handling options.

`-` stop handling options and process standard input.

lua(1)

The documentation at lua.org.

DIAGNOSTICS

Error messages should be self explanatory.

AUTHORS

R. Ierusalimschy, L. H. de Figueiredo, W. Celes

\$Date: 2011/11/16 13:53:40 \$

LUAC(1)