



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

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'jstack.1' command

\$ man jstack.1

jstack(1) Troubleshooting Tools jstack(1)

NAME

jstack - Prints Java thread stack traces for a Java process, core file, or remote debug server. This command is experimental and unsupported.

SYNOPSIS

jstack [options] pid

jstack [options] executable core

jstack [options] [server-id@] remote-hostname-or-IP

options

The command-line options. See Options.

pid The process ID for which the stack trace is printed. The process must be a Java process. To get a list of Java processes running on a machine, use the `jps(1)` command.

executable

The Java executable from which the core dump was produced.

core The core file for which the stack trace is to be printed.

remote-hostname-or-IP

The remote debug server hostname or IP address. See `jsadebugd(1)`.

server-id

An optional unique ID to use when multiple debug servers are running on the same remote host.

DESCRIPTION

The `jstack` command prints Java stack traces of Java threads for a specified Java process, core file, or remote debug server. For each Java frame, the full class name, method name, byte code index (BCI), and line number, when available, are printed. With the `-m` option, the `jstack` command prints both Java and native frames of all threads with the program counter (PC). For each native frame, the closest native symbol to PC, when available, is printed. C++ mangled names are not demangled. To demangle C++ names, the output of this command can be piped to `c++filt`. When the specified process is running on a 64-bit Java Virtual Machine, you might need to specify the `-J-d64` option, for example: `jstack -J-d64 -m pid`.

Note: This utility is unsupported and might not be available in future release of the JDK. In Windows Systems where the `dbgeng.dll` file is not present, Debugging Tools For Windows must be installed so these tools work. The `PATH` environment variable needs to contain the location of the `jvm.dll` that is used by the target process, or the location from which the crash dump file was produced. For example:

```
set PATH=<jdk>\jre\bin\client;%PATH%
```

OPTIONS

`-F`

Force a stack dump when `jstack [-l] pid` does not respond.

`-l`

Long listing. Prints additional information about locks such as a list of owned `java.util.concurrent.ownable` synchronizers. See the `AbstractOwnableSynchronizer` class description at <http://docs.oracle.com/javase/8/docs/api/java/util/concurrent/locks/AbstractOwnableSynchronizer.html>

`-m`

Prints a mixed mode stack trace that has both Java and native C/C++ frames.

`-h`

Prints a help message.

`-help`

Prints a help message.

KNOWN BUGS

In mixed mode stack trace, the `-m` option does not work with the remote debug server.

SEE ALSO

? `pstack(1)`

? `C++filt(1)`

? `jps(1)`

? `jsadebugd(1)`

JDK 8

21 November 2013

`jstack(1)`