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 'btreplay.8' command

$ man btreplay.8

BTREPLAY(8) BTREPLAY(8)

NAME

btreplay - recreate IO loads recorded by blktrace

SYNOPSIS

btreplay [options] <dev...>

DESCRIPTION

The btrecord and btreplay tools provide the ability to record and re? play IOs captured by the blktrace utility. Attempts are made to main? tain ordering, CPU mappings and time-separation of IOs.

The blktrace utility provides the ability to collect detailed traces from the kernel for each IO processed by the block IO layer. The traces provide a complete timeline for each IO processed, including detailed information concerning when an IO was first received by the block IO layer ? indicating the device, CPU number, time stamp, IO direction, sector number and IO size (number of sectors). Using this information, one is able to replay the IO again on the same machine or another set up entirely.

The basic operating work-flow to replay IOs would be something like:

-

Run blktrace to collect traces. Here you specify the device or devices that you wish to trace and later replay IOs upon.

Note:

the only traces you are interested in are QUEUE requests ?

thus, to save system resources (including storage for traces), one could specify the -a queue command line option to blktrace.

- While blktrace is running, you run the workload that you are interested in.

- When the work load has completed, you stop the blktrace utility (thus saving all traces over the complete workload).

- You extract the pertinent IO information from the traces saved by blktrace using the btrecord utility. This will parse each trace file created by blktrace, and crafty IO descriptions to be used in the next phase of the workload processing.

- Once btrecord has successfully created a series of data files to be processed, you can run the btreplay utility which attempts to generate the same IOs seen during the sample workload phase.

OPTIONS

-c <num>

--cpus=<num>

Set number of CPUs to use.

-d <dir>

--input-directory=<dir>

Set input directory. This option requires a single parameter providing the directory name for where input files are to be found. The default directory is the current directory (.).

-F

--find-records

Find record files automatically This option instructs btreplay to go find all the record files in the directory specified (ei? ther via the -d option, or in the default directory (.).

-h

--help

Show help and exit.

-i <basename>

--input-base=<basename>

Set base name for input files. Each input file has 3 fields:

1.

Device identifier (taken directly from the device name of the blktrace output file).

2.

btrecord base name ? by default ``replay''.

3.

The CPU number (again, taken directly from the blktrace output file name).

This option requires a single parameter that will override the default name (replay), and replace it with the specified value.

-I <num>

--iterations=<num>

Set number of iterations to run. This option requires a single parameter which specifies the number of times to run through the input files. The default value is 1

-M <filename>

--map-devs=<filename>

Specify device mappings. This option requires a single parame? ter which specifies the name of a file contain device mappings. The file must be very simply managed, with just two pieces of data per line:

-

The device name on the recorded system (with the '/dev/' removed). Example: /dev/sda would just be sda.

-

The device name on the replay system to use (again, without

the

'/dev/' path prepended).

An example file for when one would map devices /dev/sda and /dev/sdb on the recorded system to dev/sdg and sdh on the replay system would be:

sda sdg

sdb sdh

The only entries in the file that are allowed are these two ele? ment lines ? we do not (yet?) support the notion of blank lines, or comment lines, or the like.

The utility allows for multiple -M options to be supplied on the command line.

-N

--no-stalls

Disable pre-bunch stalls. When specified on the command line, all pre-bunch stall indicators will be ignored. IOs will be re? played without inter-bunch delays.

-x <factor>

--acc-factor=<factor>

Specify acceleration factor. Default value is 1 (no accelera? tion).

-v

--verbose

Enable verbose output. When specified on the command line, this option instructs btreplay to store information concerning each stall and IO operation performed by btreplay. The name of each file so created will be the input file name used with an exten? sion of .rep appended onto it. Thus, an input file of the name sdab.replay.3 would generate a verbose output file with the name sdab.replay.3.rep in the directory specified for input files.

In addition, btreplay will also output to stderr the names of the input files being processed.

-V

--version

Show version number and exit.

-W

--write-enable

Enable writing during replay. As a precautionary measure, by default btreplay will not process write requests. In order to enable btreplay to actually write to devices one must explicitly specify the -W option.

AUTHORS

btreplay was written by Alan D. Brunelle. This man page was created from the btreplay documentation by Bas Zoetekouw.

REPORTING BUGS

Report bugs to <linux-btrace@vger.kernel.org>

COPYRIGHT

SEE ALSO

The full documentation for btreplay can be found in /usr/share/doc/blk?trace on Debian systems.

blktrace (8), blkparse (1), btrecord (8)