NAME

irqbalance − distribute hardware interrupts across processors on a multiprocessor system

SYNOPSIS

irqbalance

DESCRIPTION

The purpose of **irqbalance** is to distribute hardware interrupts across processors on a multiprocessor system in order to increase performance.

OPTIONS

-o, --oneshot

Causes irqbalance to be run once, after which the daemon exits.

-d, --debug

Causes irqbalance to print extra debug information. Implies --foreground.

-f, --foreground

Causes irqbalance to run in the foreground (without --debug).

-j, --journal

Enables log output optimized for systemd-journal.

-p, --powerthresh=<threshold>

Set the threshold at which we attempt to move a CPU into powersave mode If more than <threshold> CPUs are more than 1 standard deviation below the average CPU softirq workload, and no CPUs are more than 1 standard deviation above (and have more than 1 IRQ assigned to them), attempt to place 1 CPU in powersave mode. In powersave mode, a CPU will not have any IRQs balanced to it, in an effort to prevent that CPU from waking up without need.

-i, --banirq=<irqnum>

Add the specified IRQ to the set of banned IRQs. irqbalance will not affect the affinity of any IRQs on the banned list, allowing them to be specified manually. This option is additive and can be specified multiple times. For example to ban IRQs 43 and 44 from balancing, use the following command line: **irqbalance --banirq=43 --banirq=44**

-m, --banmod=<module_name>

Add the specified module to the set of banned modules, similar to --banirq. irqbalance will not affect the affinity of any IRQs of given modules, allowing them to be specified manually. This option is additive and can be specified multiple times. For example to ban all IRQs of module foo and module bar from balancing, use the following command line: **irqbalance --banmod=foo --banmod=bar**

-c, --deepestcache=<integer>

This allows a user to specify the cache level at which irqbalance partitions cache domains. Specifying a deeper cache may allow a greater degree of flexibility for irqbalance to assign IRQ affinity to achieve greater performance increases, but setting a cache depth too large on some systems (specifically where all CPUs on a system share the deepest cache level), will cause irqbalance to see balancing as unnecessary. **irqbalance --deepestcache=2**

The default value for deepestcache is 2.

-l, --policyscript=<script>

When specified, the referenced script or directory will execute once for each discovered IRQ, with the sysfs device path and IRQ number passed as arguments. Note that the device path argument will point to the parent directory from which the IRQ attributes directory may be directly opened. Policy scripts specified need to be owned and executable by the user of irqbalance process, if a directory is specified, non-executable files will be skipped. The script may specify zero or more key=value pairs that will guide irqbalance in the management of that IRQ. Key=value pairs are printed by the script on stdout and will be captured and interpreted by irqbalance. Irqbalance expects a zero exit code from the provided utility. Recognized key=value pairs are:

ban=[true | false]

Directs irqbalance to exclude the passed in IRQ from balancing.

balance_level=[none | package | cache | core]

This allows a user to override the balance level of a given IRQ. By default the balance level is determined automatically based on the pci device class of the device that owns the IRQ.

numa_node=<integer>

This allows a user to override the NUMA node that sysfs indicates a given device IRQ is local to. Often, systems will not specify this information in ACPI, and as a result devices are considered equidistant from all NUMA nodes in a system. This option allows for that hardware provided information to be overridden, so that irqbalance can bias IRQ affinity for these devices toward its most local node. Note that specifying a -1 here forces irqbalance to consider an interrupt from a device to be equidistant from all nodes.

Note that, if a directory is specified rather than a regular file, all files in
the directory will be considered policy scripts, and executed on adding of an irq to a database. If such a directory is specified, scripts in the directory must additionally exit with one of the following exit codes:

0 This indicates the script has a policy for the referenced irq, and that further script processing should stop

1 This indicates that the script has no policy for the referenced irq, and that script processing should continue

2 This indicates that an error has occurred in the script, and it should be skipped (further processing to continue)

-s, --pid=<file>

Have irqbalance write its process id to the specified file. By default no pidfile is written. The written pidfile is automatically unlinked when irqbalance exits. It is ignored when used with --debug or --foreground.

-t, --interval=<time>

Set the measurement time for irqbalance. irqbalance will sleep for <time> seconds between samples of the irq load on the system cpus. Defaults to 10.

ENVIRONMENT VARIABLES

IRQBALANCE_ONESHOT

Same as --oneshot.

IRQBALANCE_DEBUG

Same as --debug.

IRQBALANCE_BANNED_CPUS

Provides a mask of CPUs which irqbalance should ignore and never assign interrupts to. If not specified, irqbalance use mask of isolated and adaptive-ticks CPUs on the system as the default value. This is a hexmask without the leading '0x'. On systems with large numbers of processors,

each group of eight hex digits is separated by a comma ’,’. i.e. ‘export IRQBALANCE_BANNED_CPUS=fc0‘ would prevent irqbalance from assigning irqs to the 7th-12th cpus (cpu6-cpu11) or ‘export IRQBALANCE_BANNED_CPUS=ff000000,00000001‘ would prevent irqbalance from assigning irqs to the 1st (cpu0) and 57th-64th cpus (cpu56-cpu63). Notes: This environment variable will be discarded, please use IRQBALANCE_BANNED_CPULIST instead. Before deleting this environment variable, Introduce a deprecation period first for the consider of compatibility.

IRQBALANCE_BANNED_CPULIST
Provides a cpulist which irqbalance should ignore and never assign interrupts to. If not specified, irqbalance use mask of isolated and adaptive-ticks CPUs on the system as the default value.

SIGNALS

SIGHUP
Forces a rescan of the available IRQs and system topology.

API

irqbalance is able to communicate via socket and return it’s current assignment tree and setup, as well as set new settings based on sent values. Socket is abstract, with a name in form of **irqbalance<PID>.sock** , where <PID> is the process ID of irqbalance instance to communicate with. Possible values to send:

stats Retrieve assignment tree of IRQs to CPUs, in recursive manner. For each CPU node in tree, it’s type, number, load and whether the save mode is active are sent. For each assigned IRQ type, it’s number, load, number of IRQs since last rebalancing and it’s class are sent. Refer to types.h file for explanation of defines.

setup Get the current value of sleep interval, mask of banned CPUs and list of banned IRQs.

settings sleep <s>
Set new value of sleep interval, <s> >= 1.

settings cpus <cpu_number1> <cpu_number2> ...
Ban listed CPUs from IRQ handling, all old values of banned CPUs are forgotten.

settings ban irqs <irq1> <irq2> ...
Ban listed IRQs from being balanced, all old values of banned IRQs are forgotten.

irqbalance checks SCM_CREDENTIALS of sender (only root user is allowed to interact). Based on chosen tools, ancillary message with credentials needs to be sent with request.

HOMEPAGE

https://github.com/Irqbalance/irqbalance