
Subject: BC: resource beancounters (v6) (with userpages reclamation + configfs)

Posted by [dev](#) on Thu, 09 Nov 2006 16:42:56 GMT

[View Forum Message](#) <> [Reply to Message](#)

MAJOR CHANGES in v6 (see details below):

- configfs interface instead of syscalls (as wanted by CKRM people...)
- added numfiles resource accounting
- added numtasks resource accounting

numfiles and numtasks controllers demonstrate how clean and simple BC interface is.

Patch set is applicable to 2.6.19-rc5-mm1

Resource BeanCounters (BC).

BC allows to account and control consumption of kernel resources used by *group* of processes (users, containers, ...).

Draft BC description on OpenVZ wiki can be found at http://wiki.openvz.org/UBC_parameters

The full BC patch set allows to control:

- kernel memory. All the kernel objects allocatable on user demand and not reclaimable should be accounted and limited for DoS protection.
e.g. page tables, task structs, vmas etc.

- virtual memory pages. BCs allow to limit a container to some amount of memory and introduces 2-level OOM killer taking into account container's consumption.
pages shared between containers are correctly charged as fractions (tunable).

- network buffers. These includes TCP/IP rcv/snd buffers, dgram snd buffers, unix, netlinks and other buffers.

- minor resources accounted/limited by number: tasks, files, flocks, ptys, siginfo, pinned dcache mem, sockets, iptentries (for containers with virtualized networking)

Summary of changes from v5 patch set:

- * configfs interface instead of syscalls (as wanted by CKRM people)
- * added numfiles resource accounting
- * added numtasks resource accounting
- * introduced dummy_resource to handle case when no resource registered
- * calls to rss accounting are integrated to rmap calls

Summary of changes from v4 patch set:

- * changed set of resources - kmemsize, privvmpages, physpages
- * added event hooks for resources (init, limit hit etc)
- * added user pages reclamation (bc_try_to_free_pages)
- * removed pages sharing accounting - charge to first user
- * task now carries only one BC pointer, simplified
- * make set_bcid syscall move arbitrary task into BC
- * resources are not recharged when task moves
- * each vm_area_struct carries a BC pointer

Summary of changes from v3 patch set:

- * Added basic user pages accounting (lockedpages/privvmpages)
- * spell in Kconfig
- * Makefile reworked
- * EXPORT_SYMBOL_GPL
- * union w/o name in struct page
- * bc_task_charge is void now
- * adjust minheld/maxheld splitted

Summary of changes from v2 patch set:

- * introduced atomic_dec_and_lock_irqsave()
- * bc_adjust_held_minmax comment
- * added __must_check for bc_*charge* funcs
- * use hash_long() instead of own one
- * bc/Kconfig is sourced from init/Kconfig now
- * introduced bcid_t type with comment from Alan Cox
- * check for barrier <= limit in sys_set_bclimit()
- * removed (bc == NULL) checks
- * replaced memcpy in beancounter_findcrate with assignment
- * moved check 'if (mask & BC_ALLOC)' out of the lock
- * removed unnecessary memset()

Summary of changes from v1 patch set:

- * CONFIG_BEANCOUNTERS is 'n' by default
- * fixed Kconfig includes in arches
- * removed hierarchical beancounters to simplify first patchset
- * removed unused 'private' pointer
- * removed unused EXPORTS

- * MAXVALUE redeclared as LONG_MAX
- * beancounter_findcreate clarification
- * renamed UBC -> BC, ub -> bc etc.
- * moved BC inheritance into copy_process
- * introduced reset_exec_bc() with proposed BUG_ON
- * removed task_bc beancounter (not used yet, for numproc)
- * fixed syscalls for sparc
- * added sys_get_bcstat(): return info that was in /proc
- * cond_syscall instead of #ifdefs

Many thanks to Oleg Nesterov, Alan Cox, Matt Helsley and others for patch review and comments.

Thanks,
Kirill
