Subject: [PATCH 1/2] sysctl: remove binary sysctls from kernel.sched_domain
Posted by Alexey Dobriyan on Mon, 06 Aug 2007 12:36:55 GMT
View Forum Message <> Reply to Message

kernel.sched_domain hierarchy is under CTL_UNNUMBERED and thus
unreachable to sysctl(2). Generating .ctl_number's in such situation is
not useful.

Signed-off-by: Alexey Dobriyan <adobriyan@sw.ru>
---

 kernel/sched.c | 38 ++++++++++++++++++++++++++----------
 1 file changed, 21 insertions(+), 17 deletions(-)

--- a/kernel/sched.c
+++ b/kernel/sched.c
@@ -5210,12 +5210,19 @@ static void migrate_dead_tasks(unsigned int dead_cpu)
 #if defined(CONFIG_SCHED_DEBUG) && defined(CONFIG_SYSCTL)

 static struct ctl_table sd_ctl_dir[] = {
- {CTL_UNNUMBERED, "sched_domain", NULL, 0, 0755, NULL, },
+ {
+ .procname = "sched_domain",
+ .mode = 0755,
+ },
   {0,},
 };

 static struct ctl_table sd_ctl_root[] = {
- {CTL_UNNUMBERED, "kernel", NULL, 0, 0755, sd_ctl_dir, },
+ {
+ .procname = "kernel",
+ .mode = 0755,
+ .child = sd_ctl_dir,
+ },
   {0,},
 };

@@ -5231,11 +5238,10 @@ static struct ctl_table *sd_alloc_ctl_entry(int n)
 }

 static void
-set_table_entry(struct ctl_table *entry, int ctl_name,
+set_table_entry(struct ctl_table *entry,
    const char *procname, void *data, int maxlen,
    mode_t mode, proc_handler *proc_handler)
 {
- entry->ctl_name = ctl_name;

```
  entry->procname = procname;
  entry->data = data;
  entry->maxlen = maxlen;
@@ -5248,28 +5254,28 @@ sd_alloc_ctl_domain_table(struct sched_domain *sd)
 {
  struct ctl_table *table = sd_alloc_ctl_entry(14);

- set_table_entry(&table[0], 1, "min_interval", &sd->min_interval,
+ set_table_entry(&table[0], "min_interval", &sd->min_interval,
    sizeof(long), 0644, proc_doulongvec_minmax);
- set_table_entry(&table[1], 2, "max_interval", &sd->max_interval,
+ set_table_entry(&table[1], "max_interval", &sd->max_interval,
    sizeof(long), 0644, proc_doulongvec_minmax);
- set_table_entry(&table[2], 3, "busy_idx", &sd->busy_idx,
+ set_table_entry(&table[2], "busy_idx", &sd->busy_idx,
    sizeof(int), 0644, proc_dointvec_minmax);
- set_table_entry(&table[3], 4, "idle_idx", &sd->idle_idx,
+ set_table_entry(&table[3], "idle_idx", &sd->idle_idx,
    sizeof(int), 0644, proc_dointvec_minmax);
- set_table_entry(&table[4], 5, "newidle_idx", &sd->newidle_idx,
+ set_table_entry(&table[4], "newidle_idx", &sd->newidle_idx,
    sizeof(int), 0644, proc_dointvec_minmax);
- set_table_entry(&table[5], 6, "wake_idx", &sd->wake_idx,
+ set_table_entry(&table[5], "wake_idx", &sd->wake_idx,
    sizeof(int), 0644, proc_dointvec_minmax);
- set_table_entry(&table[6], 7, "forkexec_idx", &sd->forkexec_idx,
+ set_table_entry(&table[6], "forkexec_idx", &sd->forkexec_idx,
    sizeof(int), 0644, proc_dointvec_minmax);
- set_table_entry(&table[7], 8, "busy_factor", &sd->busy_factor,
+ set_table_entry(&table[7], "busy_factor", &sd->busy_factor,
    sizeof(int), 0644, proc_dointvec_minmax);
- set_table_entry(&table[8], 9, "imbalance_pct", &sd->imbalance_pct,
+ set_table_entry(&table[8], "imbalance_pct", &sd->imbalance_pct,
    sizeof(int), 0644, proc_dointvec_minmax);
- set_table_entry(&table[10], 11, "cache_nice_tries",
+ set_table_entry(&table[10], "cache_nice_tries",
    &sd->cache_nice_tries,
    sizeof(int), 0644, proc_dointvec_minmax);
- set_table_entry(&table[12], 13, "flags", &sd->flags,
+ set_table_entry(&table[12], "flags", &sd->flags,
    sizeof(int), 0644, proc_dointvec_minmax);

  return table;
@@ -5289,7 +5295,6 @@ static ctl_table *sd_alloc_ctl_cpu_table(int cpu)
  i = 0;
  for_each_domain(cpu, sd) {
   snprintf(buf, 32, "domain%d", i);
-  entry->ctl_name = i + 1;
```

```
   entry->procname = kstrdup(buf, GFP_KERNEL);
   entry->mode = 0755;
   entry->child = sd_alloc_ctl_domain_table(sd);
@@ -5310,7 +5315,6 @@ static void init_sched_domain_sysctl(void)

  for (i = 0; i < cpu_num; i++, entry++) {
   snprintf(buf, 32, "cpu%d", i);
-  entry->ctl_name = i + 1;
   entry->procname = kstrdup(buf, GFP_KERNEL);
   entry->mode = 0755;
   entry->child = sd_alloc_ctl_cpu_table(i);
```

Subject: Re: [PATCH 1/2] sysctl: remove binary sysctls from kernel.sched_domain
Posted by Ingo Molnar on Mon, 06 Aug 2007 15:13:49 GMT
View Forum Message <> Reply to Message

* Alexey Dobriyan <adobriyan@sw.ru> wrote:

> kernel.sched_domain hierarchy is under CTL_UNNUMBERED and thus
> unreachable to sysctl(2). Generating .ctl_number's in such situation
> is not useful.

thanks - i've applied your patch to my tree.

 Ingo
