Subject: [PATCH] Cleanup oops/bug reports on i386
Posted by Pavel Emelianov on Wed, 08 Aug 2007 14:16:54 GMT
View Forum Message <> Reply to Message

Typically the oops first lines look like this:

BUG: unable to handle kernel NULL pointer dereference at virtual address 00000000
 printing eip:
c049dfbd
*pde = 00000000
Oops: 0002 [#1]
PREEMPT SMP
...

Such output is gained with some ugly if (!nl) printk("\n"); code and
besides being a waste of lines, this is also annoying to read. The
following output looks better (and it is how it looks on x86_64):

BUG: unable to handle kernel NULL pointer dereference at virtual address 00000000
printing eip: c049dfbd *pde = 00000000
Oops: 0002 [#1] PREEMPT SMP
...

Signed-off-by: Pavel Emelyanov <xemul@openvz.org>

---

 arch/i386/kernel/traps.c | 16 ++++----------
 arch/i386/mm/fault.c | 13 +++++++----
 2 files changed, 11 insertions(+), 18 deletions(-)

diff --git a/arch/i386/kernel/traps.c b/arch/i386/kernel/traps.c
index a20c7ad..f469d20 100644
--- a/arch/i386/kernel/traps.c
+++ b/arch/i386/kernel/traps.c
@@ -368,31 +368,23 @@ void die(const char * str, struct pt_reg
   local_save_flags(flags);

  if (++die.lock_owner_depth < 3) {
- int nl = 0;
   unsigned long esp;
   unsigned short ss;

   report_bug(regs->eip, regs);

- printk(KERN_EMERG "%s: %04lx [#%d]\n", str, err & 0xffff, ++die_counter);
+ printk(KERN_EMERG "%s: %04lx [#%d] ", str, err & 0xffff, ++die_counter);
 #ifdef CONFIG_PREEMPT

```
-  printk(KERN_EMERG "PREEMPT ");
-  nl = 1;
+  printk("PREEMPT ");
 #endif
 #ifdef CONFIG_SMP
-  if (!nl)
-   printk(KERN_EMERG);
   printk("SMP ");
-  nl = 1;
 #endif
 #ifdef CONFIG_DEBUG_PAGEALLOC
-  if (!nl)
-   printk(KERN_EMERG);
   printk("DEBUG_PAGEALLOC");
-  nl = 1;
 #endif
-  if (nl)
-   printk("\n");
+  printk("\n");
+
   if (notify_die(DIE_OOPS, str, regs, err,
      current->thread.trap_no, SIGSEGV) !=
     NOTIFY_STOP) {
diff --git a/arch/i386/mm/fault.c b/arch/i386/mm/fault.c
index fece684..93263aa 100644
--- a/arch/i386/mm/fault.c
+++ b/arch/i386/mm/fault.c
@@ -537,23 +537,22 @@ no_context:
    printk(KERN_ALERT "BUG: unable to handle kernel paging"
     " request");
   printk(" at virtual address %08lx\n",address);
-  printk(KERN_ALERT " printing eip:\n");
-  printk("%08lx\n", regs->eip);
+  printk(KERN_ALERT "printing eip: %08lx ", regs->eip);

   page = read_cr3();
   page = ((__typeof__(page) *) __va(page))[address >> PGDIR_SHIFT];
 #ifdef CONFIG_X86_PAE
-  printk(KERN_ALERT "*pdpt = %016Lx\n", page);
+  printk("*pdpt = %016Lx ", page);
   if ((page >> PAGE_SHIFT) < max_low_pfn
       && page & _PAGE_PRESENT) {
    page &= PAGE_MASK;
    page = ((__typeof__(page) *) __va(page))[(address >> PMD_SHIFT)
                                             & (PTRS_PER_PMD - 1)];
-   printk(KERN_ALERT "*pde = %016Lx\n", page);
+   printk(KERN_ALERT "*pde = %016Lx ", page);
    page &= ~_PAGE_NX;
```

```
   }
 #else
-  printk(KERN_ALERT "*pde = %08lx\n", page);
+  printk("*pde = %08lx ", page);
 #endif

   /*
@@ -567,8 +566,10 @@ no_context:
    page &= PAGE_MASK;
    page = ((__typeof__(page) *) __va(page))[(address >> PAGE_SHIFT)
                                             & (PTRS_PER_PTE - 1)];
-   printk(KERN_ALERT "*pte = %0*Lx\n", sizeof(page)*2, (u64)page);
+   printk("*pte = %0*Lx ", sizeof(page)*2, (u64)page);
   }
+
+  printk("\n");
  }

  tsk->thread.cr2 = address;
```