Michal Ludvig
<michal@logix.cz>

- Home Page
- Personal Page
- Humorník
- Nový Zéland
- Fan klub NZ
- Douglas Adams
- ©árka
- Svatba
- Wedding
- Feedback

Programming and development

- Nagios and SNMP scripts
- SMTP client
- DDNS updater
- YubiKey-LDAP
- VIA PadLock
- VIA VT-310DP pipeline configurator
- OpenChrome for OpenSUSE
- IPsec-tools
- CryptoDev 4 Linux
- libfaketime.so
- FastCrypt driver
- S/MIME decoder
- HTML lowercaser
- CygProfiler suite
- AMD64 registers
- CFI for GAS
- NetShips
- ptrace() demo
- XFree86 support for GDB
- Etc.

Publications and documents

- Linux on AMD64

Prev: **10.3 Switching to Protected Mode**
Next: **10.5 Initialization Example**

10.4 Software Initialization for Protected Mode

Most of the initialization needed for protected mode can be done either before or after switching to protected mode. If done in protected mode, however, the initialization procedures must not use protected-mode features that are not yet initialized.

10.4.1 Interrupt Descriptor Table

The IDTR may be loaded in either real-address or protected mode. However, the format of the interrupt table for protected mode is differe than that for real-address mode. It is not possible to change to protecte mode and change interrupt table formats at the same time; therefore, i inevitable that, if IDTR selects an interrupt table, it will have the wrong format at some time. An interrupt or exception that occurs at this time have unpredictable results. To avoid this unpredictability, interrupts sho remain disabled until interrupt handlers are in place and a valid IDT has been created in protected mode.

10.4.2 Stack

The SS register may be loaded in either real-address mode or protectec mode. If loaded in real-address mode, SS continues to point to the same linear base-address after the switch to protected mode.

10.4.3 Global Descriptor Table

Before any segment register is changed in protected mode, the GDT register must point to a valid GDT. Initialization of the GDT and GDTR m be done in real-address mode. The GDT (as well as LDTs) should reside RAM, because the processor modifies the accessed bit of descriptors.

10.4.4 Page Tables

Page tables and the PDBR in CR3 can be initialized in either real-addres mode or in protected mode; however, the paging enabled (PG) bit of CR cannot be set until the processor is in protected mode. PG may be set simultaneously with PE, or later. When PG is set, the PDBR in CR3 shoul already be initialized with a physical address that points to a valid page directory. The initialization procedure should adopt one of the following strategies to ensure consistent addressing before and after paging is enabled:

- The page that is currently being executed should map to the sam physical addresses both before and after PG is set.
- A JMP instruction should immediately follow the setting of PG.

10.4.5 First Task

The initialization procedure can run awhile in protected mode without initializing the task register; however, before the first task switch, the following conditions must prevail:

- There must be a valid task state segment (TSS) for the new task. stack pointers in the TSS for privilege levels numerically less than equal to the initial CPL must point to valid stack segments.
- The task register must point to an area in which to save the curre

- VIA PadLock - Wicked fast encryption
- VIA PadLock - Ďábelsky rychlé ąifrování
- Jak funguje initramdisk
- Linux a 64 bitů
- Secure networking
- Napiąte si debugger
- AMD64 - AMD Opteron
- IPv6 krok za krokem I
- IPv6 krok za krokem II
- IPv6 krok za krokem III
- Sharp Zaurus
- Mosix - počítejte rychleji! I
- Mosix - počítejte rychleji! II
- Mosix - počítejte rychleji! III
- What's new in GDB 6.0
- i386 Programmer's Manual

task state. After the first task switch, the information dumped in t area is not needed, and the area can be used for other purposes.

Prev: **10.3 Switching to Protected Mode**
Next: **10.5 Initialization Example**