Архитектура AMD64.

Константин Белоусов
kib@freebsd.org

25 сентября 2010 г.

Revision : 1.8

Common Wisdom

- 64-битная машина
- 64-битное адресное пространство
- обратно совместима с i386

AMD64

Common Wisdom

- 64-битная машина
- 64-битное адресное пространство
- обратно совместима с i386

На самом деле

- 64-битная машина, но 32-битный поток инструкций
- 64-битное адресное пространство теоретически
- обратно совместима с i386, но не полностью

32-битный поток инструкций

`movq offset(%rax),%rbx` ; *offset* - signed 32bit

32-битный поток инструкций

`movq offset(%rax),%rbx` ; *offset* - signed 32bit

The only instruction to load data from the non-based 64bit address
`movq address,%rax`

32-битный поток инструкций

`movq offset(%rax),%rbx` ; *offset* - signed 32bit

The only instruction to load data from the non-based 64bit address
`movq address,%rax`

NOP

- amd64 zero-extends 32bit operations
- `nop == .byte 0x90 == xchgl %eax,%eax in 32bit mode`
- `xchgl %eax,%eax == .byte 0x87, 0xc0`

Memory model

-mcmodel=

- small: текст и данные в младших 2Gb
- medium: текст в младших 2Gb
- kernel: текст в старших 2Gb
- large: нет ограничений

Compiler

Memory model

-mcmodel=

- small: текст и данные в младших 2Gb
- medium: текст в младших 2Gb
- kernel: текст в старших 2Gb
- large: нет ограничений

-fPIC

pc-relative addressing

Адресное пространство

Таблицы страниц

- 48 бит физического адреса, $2^{48} = 65536\,GiB = 64\,TiB$
- 52 бита виртуального адреса, $2^{52} = 1048576\,GiB = 1024\,TiB$
- 4x-уровневые таблицы
- PAE

Совместимость

CPU modes

- Long mode (64 бита)
- 32bit (и protected 16bit)
- vm86 нет

COMPAT_FREEBSD32

Зачем ?

- i386 - самая популярная архитектура
- Иногда i386 на 20% быстрее
- 4GiB адресного пространства вместо 3GiB
- NX bit

COMPAT_FREEBSD32

Зачем ?

- i386 - самая популярная архитектура
- Иногда i386 на 20% быстрее
- 4GiB адресного пространства вместо 3GiB
- NX bit

Когда быстрее ?

- интенсивная работа со сложными структурами данных
- Core2 не умеет macrofusion в LM

Зачем ? 32-битные приложения

- Wine
- Binary win32 codecs
- Very old 5.x libkse *Никогда не пользуйтесь статической линковкой*

Segmentation

Зачем ? 32-битные приложения

- Wine
- Binary win32 codecs
- Very old 5.x libkse *Никогда не пользуйтесь статической линковкой*

Сегментация в 64-битном режиме

- VMWare. Использовала segment limits.
- MSRC000_0080 Extended Feature Enable Register (EFER), LMSLE(13): long mode segment limit enable. Read-write. 1=Enables the long mode segment limit check mechanism.

cc -m32

- compiler and toolchain
- /usr/lib32
- csu
- /usr/include/machine

Solaris model

- Ядро – 64bit
- Библиотеки – 64bit & 32bit
- Программы – почти все 32bit
- Программы 64bit: отладчик, procfs(5) utilities – работающие с ABI других процессов
- isaexec