

Linuxovy kernel

Pavel Machek
<pavel@ucw.cz>

Co je cilem

pripravit prijemne prostredi pro uzivatelske procesy

- schovat rozdily mezi ruznymi pocitaci
- schovat rozdil mezi RAM a diskem
- oddelit od sebe procesy
- umoznit sdileni zdroju co jsou tam jednou (disk, sit)

bezpecnost

- ochranit same sebe
- oddelit od sebe uzivatele

Co neni cilem

sprava tisku

binarni kompatibilita mezi ruznymi pocitaci

oddeleni/ochrana kusu jadra mezi sebou

Proc je jadro dulezite

schovava rozdily mezi jednotlivymi pocitaci

- preportujete jadro, a muzete nainstalovat Debian
- teda skoro, pak je tu jeste maly detail s grafikou
- ...a tiskarnou
- ...a cimkoliv co neni "bezne"

Co je v jadre

"zaklad"

sprava pameti

scheduler, prepinani procesu

filesystemy

...a co je nejdulezitejsi?

drivery, spousty driveru

Jadro je zle

Zadny debugger

- ...jen dmesg
- ...a backtracy

Chyba muze spůsobit cokoliv

- ...a projevovat se jen u cizich

Jadro bezi na vice procesorech najednou

- ...zamky, RCU
- ...cache, DMA

Kusy jsou v assembleru

Vyvojari jadra jsou hodni

...dokud se nenastvou

POSIX -- je jasne co to ma delat

Ccko

Slusne napsany kod

Dokumentace, dokonce i v cinstine

Některé věci se jmenují jinak

printf -> printk

malloc/free -> kmalloc(GFP_...),kfree

A maji silne hracky

kernelove moduly

qemu

hardwarove debuggery (JTAG)

Architektury

Cross-kompilace

Konfigurace jadra

Zle veci slozite

initrd

systemd

X11

udev

Interface na userland

syscall-y

/dev/*

/proc/*

/sys/*

Jak neprogramovat do jadra

iopl()

/dev/uio

usb

Moduly

Nejlepší je neprogramovat vůbec

Device tree

Nejlepší je neprogramovat vůbec

Device tree