

WIP: Speeding up FreeBSD booting

From `mi_startup` to `moutrout`

Colin Percival
Tarsnap Backup Inc.
`cperciva@tarsnap.com`

September 21, 2017

FreeBSD boot process

- BIOS / EFI
- FreeBSD boot loader(s)
- FreeBSD kernel initialization
 - `mi_startup` runs SYSINITs
 - Ends with `start_init` mounting / and spawning `init`
- FreeBSD userland initialization
 - `rc.d` scripts

FreeBSD boot process

- BIOS / EFI
- FreeBSD boot loader(s)
- FreeBSD kernel initialization
 - `mi_startup` runs SYSINITs ← Let's make this faster.
 - Ends with `start_init` mounting / and spawning `init`
- FreeBSD userland initialization
 - `rc.d` scripts

What's taking so long?

- 0.7s spent initializing `vm_page` array (≈ 20 ms / GB RAM)
 - markj committed rewrite in r323290; 3x speedup
 - Can we do better? Initialize 1 GB of pages, then come back for the rest later?
- 1.0s spent calibrating the TSC
- 1.0s spent calibrating the local APIC timer
 - Code is literally “read counter; DELAY(1000000); read counter”
 - WIP yields 50x speedup w/o loss of accuracy, hope to commit soon
- 1.5s spent drawing text on the screen
 - Scrolling results in entire screen being redrawn
 - Lots of time is spent drawing blanks on top of blanks
 - WIP yields 5x speedup, hope to commit soon

What's taking so long?

- 0.5s spent initializing `kbd0`
- 2.0s spent initializing `psm0`
 - Mostly time spent DELAYing after sending reset commands
 - Help wanted: Does anyone know why we wait so long?
- 4.0s spent at `Root mount waiting for usb0`
 - Workaround via `hw.usb.no_boot_wait=1`
 - Help wanted: Can we detect if the devices we need are available and skip this automatically?
- < 1.0s spent on everything else

- I hope to reduce kernel init time on my laptop from ≈ 11.5 seconds to ≈ 1.0 seconds.
- ... hopefully in time to give a talk at BSDCan.

- I hope to reduce kernel init time on my laptop from ≈ 11.5 seconds to ≈ 1.0 seconds.
- ... hopefully in time to give a talk at BSDCan.

Questions?