



What can Linux learn from the others

Introducing interesting Solaris features

Stanislav Kozina

Software Maintenance Engineer, Red Hat

17/02/2012

Disclaimer

“The following presentation contains my personal opinions and thoughts and in no way does express official statements of my recent, past or future employers.”



Talk Goals

- Present Solaris as
 - Actively developed system
 - Opensource system
 - Interesting feature-rich platform



Solaris History

- 1983: 4.1BSD -> SunOS 1.0
- 1992: SVR4 -> SunOS 5.x, Solaris 2
- 2005: OpenSolaris, CDDL license
- 2009: Oracle buys Sun
- 2010: OpenSolaris cancelled
- 2010: Project Illumos started
- 2011: Solaris 11 under proprietary license



Some Legacy Features

- LWP – Leighweighted processes
- STREAMS
- DLPI/XPI – DataLink Provider Interface



Kernel Stability

- Driven by enterprise requirements
- Each documented function has stability level
- Binary compatibility ensured for a long time
- Stability(5)



Illumos

- Based on last OpenSolaris bits
- Kernel developed under CDDL license
- Several distributions
 - OpenIndiana
 - SmartOS
 - Nexenta
- Gnome2, OpenOffice
- Troubles with wireless networks



Interesting Features

- ZFS
- Dtrace
- MDB/KMDB
- Crossbow (network virtualization)
- Zones
- RBAC
- FMA



Virtual networking

- Projects Crossbow, Nemo
- Virtual network card – driver instance
- Shaping
- Completely separated network stacks



ZFS

- FS + Logical/Physical Volume manager
- B-tree, Copy-on-Write
- Simple to use – zpool, zfs commands
- Very safe - checksums
- Used for system updates



DTrace

- Dynamic Tracing toolkit
- Intermediate language with sanity checks
- Work on each system out-of-box
- Non-root usage
- Userspace tracing, Java tracing...
- Chill(), panic()
- Fast (HelloWorld in 0.6 sec)



Compressed Type Format (CTF)

- Function prototypes & Type definitions
- Really compressed (~100KB per kernel)
- SUNW_ctf section in ELF header
- Merged during linkage phase
- Used by DTrace (fbt provider) and MDB



MDB/KMDB

- Debugger (kernel focused) and CDA tool
- Usage on live systems as well as cores
 - Print structures and data
 - Stop on given instruction
- Quite uncommon syntax



Crash dumps

- Created automatically
- On panic crash dump is saved to device
 - UFS – swap partition
 - ZFS – dedicated swap/dump partition
- On reboot can be extracted on saved elsewhere



kmem_flags

- Tunables on kernel SLAB allocator
 - TEST – check free buffers with patterns
 - REDZONE – Check writes after buffer end
 - AUDIT – Log of most recent thread stacks
- Great for debugging multiple threads



Thanks for attention!

