Linux Kernel

Vitaly Karasik, RHCE
vitaly _at_ karasik.org
TELUX, 3.2006

Agenda

- What is Linux & Linux kernel?
- How many kernels we have?
- Kernel modules
- Kernel boot-time parameters
- Tuning kernel using sysctl (/proc)
- limits.conf
- Upgrading Linux kernel
- Do we need recompile kernel?
- Demo

Linux is ...

Linux is :

- kernel
- system tools (boot loader, etc.)
- shell, user tools (util-linux, etc.)
- system libraries, compilers, etc.
- daemons (DNS, WEB, mail, etc)
- applications (office, CAD/CAM, etc)

Linux Kernel history

- 08.1991 - "I'm doing a (free) operating system (just a hobby) for 386(486) AT clones" Linus (torvalds@kruuna.helsinki.fi)
- 1995 - 1.2 kernel (kernel modules, firewall, etc)
- 1996 - 2.0 kernel (multiple platform, SMP support)
- 1999 - 2.2 kernel (64-bit, etc)
- 2001 - 2.4 kernel
- 12.2003 - 2.6 kernel

2.4 kernel

- stateful firewall (Netfilter/iptables)
- better RAID support
- big RAM, big filesystems/files
- NFSv3, improved SMB
- better PCMCIA, USB, ISA PnP support

The wonderful world of Linux 2.6

- better support for small and big system
- udev - dynamic /dev
- hyperthreading and 32_64 support
- many system limits have been increased (1 billion PIDs, more disks, users)
- new scheduler
- NPTL (Native POSIX Thread Library)
- HW support (SATA, USB, WIFI, ...)

A Forest of Kernel Trees (by Moshe Bar)

- 2.0 tree
  - 2.2 tree
  - 2.4 tree
    - standart 2.4
    -ac (Alan Cox)
    -aa (Andrea Arcangeli)
    -mjc (Michael Cohen)
    - 2.4 RedHat kernel, 2.4 RHEL kernel
  - 2.6 tree

finger @finger.kernel.org

Kernel modules

Kernel = kernel (/boot/vmlinuz) + modules (/lib/modules/`uname -r`)

- /etc/modules.conf | modprobe.conf , modules options

Utilities:

- lsmod
- insmod, modprobe
- rmmod
- modinfo
- initrd, mkinitrd

Kernel boot-time parameters

- /usr/share/doc/kernel-doc-2.x.y/kernel-parameters.txt

- examples:
    - root=/dev/hda2 - root filesystem
    - mem=256m - force usage of a specific amount of memory
    - quiet - disable log messages

The /proc filesystem

- Read-only access
  - /proc/cpuinfo
  - /proc/meminfo
- Read/Write access
  - /proc/sys/fs/file-max - max number of file handlers
  - /proc/sys/net/ipv4/ip_forward - IP forwarding
- Access to processes internals
  - /proc/*nnnn*

The /proc filesystem & sysctl

- sysctl -a - complete list of /proc/sys params
- sysctl -p - Load in sysctl settings from /etc/sysctl.conf
- /etc/sysctl.conf - sysctl config file
- With /proc fs we can
  - get real-time information about kernel and processes
  - tuning VM (virtual memory)
  - tuning IPC (inter-process communications)
- tuning network params for security & performance

limits.conf

- part of PAM
- ulimit -a - get user limits
- setrlimit(), getrlimit() - get/set resource limits and usage
- Examples:
    - ulimit -c - core file size
    - ulimit -n - max open files
- soft & hard limits

- /etc/security/limits.conf
- /usr/share/doc/pam-0.xy/txts/README.pam_limits

Kernel binary upgrade

rpm -iv kernel-2.x.y.-z.i686.rpm

It's all!!!

Do you need kernel recompilation? - NO

No, because:

- we receive several tested and optimized kernels from distributor (regular, SMP, bigmem, i386, i686, ...)
- we can tune kernel with boot-time params, sysctl, initrd and limits.conf
- we can add/replace kernel modules without kernel recompilation (ntfs, lucent modem, nvidia, ...)
- RH doesn't support custom kernel

Do you need kernel recompilation? - Yes

Yes, because:

- We need some beta/unstable features in linux kernel
- We need patched kernel
- We need monolite kernel
- We have small RAM
- We're kernel hackers
- It's nice stress test for CPU/RAM/disks/network

Demo

- kernel binary upgrade
- kernel module install from binary rpm
- kernel module install from source
- initrd internals
- kernel compilation

Resources

- www.kernel.org
- www.linuxdoc.org - Kernel[-build] HOWTO
- www.linuxhq.com
- www.kernelnewbies.org
- lwn.net/KernelPatches
 - /proc fs - RH Linux Reference Guide, proc.txt from kernel sources

Resources - II

- under /usr/share/doc/kernel-doc-2.6.x/
  - sysrq.txt - "Magic keys"
    - proc.txt - about /proc fs
    - ip-sysctl.txt - networking
    - vm.txt - virtual memory

Resources - III

Tuning

- people.redhat.com/alikins/system_tuning.html
- samag.com - Linux Kernel Tunign using System Control
- IBM Developerworks - Administer Linux on the fly (with good links collection)
- http://publib-b.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/redp3861.html?Open - "Tuning Red Hat Enterprise Linux on xSeries Servers"

http://linux.dell.com/dkms/dkms.html Dynamic Kernel Module Support.

Resources - IV

2.6

- www.kniggit.net/wwol26.html - the wonderful world of Linux 2.6

- http://kerneltrap.org/node/799?PHPSESSID=24bdf202b80751145248c5684117f650 HowTo Upgrade To The 2.6 Kernel

- http://linuxdevices.com/articles/AT5793467888.html Migrating to Linux kernel 2.6:
Using the 2.6 kernel with your current system, Customizing a 2.6-based Linux kernel, Migrating device drivers to Linux kernel 2.6,
Migrating custom Linux installations to 2.6, Migrating applications to the 2.6 kernel and NPTL

- http://lwn.net/Articles/162686/ - Linux in a binary world... a doomsday scenario (about binary-only kernel modules)