Creating custom initrd images

Dag Wieërs
<dag@wieers.com>

Dag Who ?

- Independent Linux consultant
- Worked for IBM, Euroclear, EMC, AXA, ...
- Open Source projects:
  - RPMforge and ELRepo RPM packages
  - Ex-CentOS team member
  - Dstat, WiiPresent, proxytunnel, unoconv, mrepo, ...

Objective

- After the presentation you should be able to:
  - Understand the difference between initrd and initramfs
  - Uncompress and examine them
  - Construct your own with minimal effort
  - Do funky new stuff with them

What is an initrd ?

- Since the beginning of (Linux v1.xx) time
- There to:
  - Facilitate loading additional modules
  - Prepare the system to “find” the root file system
  - Then bootstraps this root file system
- But, of course can be used as the definitive system

What is there to prepare ?

- To reduce the kernel size and have more flexibility, lots of important drivers are in loadable modules
  - Hardware drivers for block devices
  - LVM
  - Software RAID
  - Disk encryption
  - Booting from network
  - Filesystem drivers
- An initrd can be made specifically for your hardware, during install or when updating the kernel (eg. using the /etc/modprobe.conf)

initrd vs initramfs

- Differently allocated in the kernel
  - initrd is loaded into a statically allocated RAM disk
  - initramfs uses a dynamically allocated RAM disk
- Differently driven
  - initrd needs a special file system driver
  - initramfs uses lightweight ramfs driver (built-in)
- Differently bootstrapped
  - initrd kicks off /linuxrc
  - initramfs initiates /init
- Initrd predates initramfs
  - initrd is from Linux kernel 2.4 and older
  - initramfs since Linux kernel 2.6

Dissection of an “initrd”

- Consists of a (compressed) cpio archive
  - Can also be a filesystem (if the kernel can read it)
- Can be easily inspected by doing:
  - gzip -dc initrd.img | cpio -idv
- /linuxrc and /init contain the script that is executed
  - This is usually a shell script (eg. busybox /bin/sh or Red Hat's /bin/nash)
  - In principal it doesn't matter what interpreter

“initrd” in distributions

- Every distribution has a framework to recreate the used initrd or initramfs to update drivers or enable/disable functionality:
  - Red Hat: mkinitrd
  - Debian: mkinitramfs
  - By hand:
    - find . | cpio –quiet -c -o | gzip > initrd.img

Example project: AXAdebug

- AXAdebug
  - Recurring problems with network connectivity
  - Need for testing network patches by technicians
  - End-to-end testing of basic functionality
  - Causes:
    - Wrong interfaces used
    - Wrong switch-ports used
    - Configuration on switch missing/wrong
    - Firewall rules blocking basic protocols for loading
- Demonstration (unprepared ! will be fun)

Example tool: siplinux

- Goal: minimal rescue image
  - RIPLinux at one point was pronounced death by the author
  - I wanted a complete rescue image based on CentOS
- Result:
  - “Recovery Is Possible” becomes “Salvation Is Possible”
  - Automated using mkrootfs.sh

What does mkrootfs do ?

- It creates a initramfs based on a set of binaries and template files
- It populates the necessary files to make it boot
- So you can use it to create minimal custom initramfs

Live demonstration

- Unprepared (even more fun !)

More information

- Google please :-)