



Home | News | Articles | Polls | Forum

Keywords: Match: **All keywords** Search articles

Introducing initramps, a new model for initial RAM disks

by Rob Landley, TimeSys (Mar. 15, 2005)



Foreword: This technical article introduces initramps, a Linux 2.6 feature that enables an initial root filesystem and init program to reside in kernel memory cache, rather than on a ramdisk, as with initrd filesystems. Compared to initrd, initramps can increase boot-time flexibility, memory efficiency, and simplicity, the author says.

One especially interesting feature for embedded Linux developers is that relatively simple, deeply embedded systems can use initramps as their sole filesystem.

This article was initially published in the company's [monthly newsletter](#). Enjoy . . . !

Introducing initramps, a new model for initial RAM disks

The problem. (Why "root=" doesn't scale.)

When the Linux kernel boots the system, it must find and run the first user program, generally called "init". User programs live in filesystems, so the Linux kernel must find and mount the first (or "root") filesystem in order to boot successfully.

Ordinarily, available filesystems are listed in the file /etc/fstab so the mount program can find them. But /etc/fstab is itself a file, stored in a filesystem. Finding the very first filesystem is a chicken and egg problem, and to solve it the kernel developers created the kernel command line option "root=", to specify which device the root filesystem lives on.

Fifteen years ago, "root=" was easy to interpret. It was either a floppy drive or a partition on a hard drive. These days the root filesystem could be on dozens of different types of hardware (SCSI, SATA, flash MTD), or even spread across several of them in a RAID. Its location could move around from boot to boot, such as hot pluggable USB devices on a system with multiple USB ports -- when there are several USB devices, which one is correct? The root filesystem might be compressed (how?), encrypted (with what keys?), or loopback mounted (where?). It could even live out on a network server, requiring the kernel to acquire a DHCP address, perform a DNS lookup, and log in to a remote server (with username and password), all before the kernel can find and run the first userspace program.

These days, "root=" just isn't enough information. Even hard-wiring tons of special case behavior into the kernel doesn't help with device enumeration, encryption keys, or network logins that vary from system to system. Worse, programming the kernel to perform these kind of complicated multipart tasks is like writing web software in assembly language: it can be done, but it's considerably easier to simply use the proper tools for the job. The kernel is designed to follow orders, not give them.

With no end to this ever-increasing complexity in sight, the kernel developers decided to back up and find a better way to deal with the whole problem.

The solution

Linux 2.6 kernels bundle a small ram-based initial root filesystem into the kernel, and if this filesystem contains a program called "init" the kernel runs that as its first program. At that point, finding some other filesystem containing some other program to run is no longer the kernel's problem, but is now the job of the new program.

The contents of initramps don't have to be general purpose. If a given system's root filesystem lives on an encrypted network block device, and the network address, login, and decryption key are all to be found on a USB device named "larry" (which requires a password to access), that system's initramps can have a special-purpose program that knows all about that, and makes it happen.

For systems that don't need a large root filesystem, there's no need to locate or switch to any other root filesystem.

How is this different from initrd?

The linux kernel already had a way to provide a ram-based root filesystem, the initrd

Got a HOT tip? [please tell us!](#)

Free weekly newsletter
Enter your email...
Subscribe

Click for a profile of each sponsor:

SUPER-PLATINUM SPONSOR

moblin.org

MOBLIN NEWS & LINKS
 • [Moblin Official Blog](#)
 • [Moblin to Go](#)
 • [Wind River's Moblin stack](#)
 • [Adobe AIR for devices](#)

FEATURED VIDEO

 Moblin v2 "Fastboot"

PLATINUM SPONSORS

timesys | **WIND RIVER**

Qt Code less. Create more. Deploy everywhere.

montavista

NOMACHINE
THE NETWORK COMPUTING COMPANY

GOLD SPONSORS

infotek

[\(Become a sponsor\)](#)

ADVERTISEMENT

VYATTA
Open Source Networking

"...achieves 2x the throughput of a Cisco 2821 at 1/2 the price"
-Tolly Group

FREE DOWNLOAD

[\(Advertise here\)](#)

Check out the latest Linux powered...



Resource Library

mechanism. For 2.4 and earlier kernels, initrd is still the only way to do this sort of thing. But the kernel developers chose to implement a new mechanism in 2.6 for several reasons.

ramdisk vs ramfs

A ramdisk (like initrd) is a ram based block device, which means it's a fixed size chunk of memory that can be formatted and mounted like a disk. This means the contents of the ramdisk have to be formatted and prepared with special tools (such as mke2fs and losetup), and like all block devices it requires a filesystem driver to interpret the data at runtime. This also imposes an artificial size limit that either wastes space (if the ramdisk isn't full, the extra memory it takes up still can't be used for anything else) or limits capacity (if the ramdisk fills up but other memory is still free, you can't expand it without reformatting it).

But ramdisks actually waste even more memory due to caching. Linux is designed to cache all files and directory entries read from or written to block devices, so Linux copies data to and from the ramdisk into the "page cache" (for file data), and the "dentry cache" (for directory entries). The downside of the ramdisk pretending to be a block device is it gets treated like a block device.

A few years ago, Linus Torvalds had a neat idea: what if Linux's cache could be mounted like a filesystem? Just keep the files in cache and never get rid of them until they're deleted or the system reboots? Linus wrote a tiny wrapper around the cache called "ramfs", and other kernel developers created an improved version called "tmpfs" (which can write the data to swap space, and limit the size of a given mount point so it fills up before consuming all available memory). Initramfs is an instance of tmpfs.

These ram based filesystems automatically grow or shrink to fit the size of the data they contain. Adding files to a ramfs (or extending existing files) automatically allocates more memory, and deleting or truncating files frees that memory. There's no duplication between block device and cache, because there's no block device. The copy in the cache is the only copy of the data. Best of all, this isn't new code but a new application for the existing Linux caching code, which means it adds almost no size, is very simple, and is based on extremely well tested infrastructure.

A system using initramfs as its root filesystem doesn't even need a single filesystem driver built into the kernel, because there are no block devices to interpret as filesystems. Just files living in memory.

Initrd vs initramfs

The change in underlying infrastructure was a reason for the kernel developers to create a new implementation, but while they were at it they cleaned up a lot of bad behavior and assumptions.

Initrd was designed as front-end to the old "root=" root device detection code, not a replacement for it. It ran a program called "/linuxrc" which was intended to perform setup functions (like logging on to the network, determining which of several devices contained the root partition, or associating a loopback device with a file), tell the kernel which block device contained the real root device (by writing the `de_t` number to `/proc/sys/kernel/real-root-dev`), and then return to the kernel so the kernel could mount the real root device and execute the real init program.

This assumed that the "real root device" was a block device rather than a network share, and also assumed that initrd wasn't itself going to be the real root filesystem. The kernel didn't even execute "/linuxrc" as the special process ID 1, because that process ID (and its special properties like being the only process that can not be killed with "kill -9") was reserved for init, which the kernel was waiting to run after it mounted the real root filesystem.

With initramfs, the kernel developers removed all these assumptions. Once the kernel launches "/init" out of initramfs, the kernel is done making decisions and can go back to following orders. With initramfs, the kernel doesn't care where the real root filesystem is (it's initramfs until further notice), and the "/init" program from initramfs is run as a real init, with PID 1. (If initramfs's init needs to hand that special Process ID off to another program, it can use the `exec()` syscall just like everybody else.)

Summary

The traditional `root=` kernel command-line option is still supported and usable, but new developments in the types of initial RAM disks supported by the kernel provide many optimizations and much-needed flexibility for the future of the Linux kernel. The next article in this series, available in next month's issue of TimeSource, explains how you can start making the transition to the new initramfs initial RAM disk mechanism.

Copyright (c) 2006, TimeSys Corp. All rights reserved. Reproduced by LinuxDevices.com with permission. This article was initially published in the [TimeSys monthly newsletter](#).



About the author: Rob Landlely is a Senior Linux Engineer at TimeSys Corporation and the new maintainer of BusyBox. He first encountered Linux when the SLS disks came across Fidonet in 1993, and has been building various Linux systems from source code for six years. He co-founded a combination Linux Expo and Science Fiction convention (Penguicon) because that really is his idea of fun, as are writing documentation, researching computer history, urban hiking, and knitting chain mail.

- [Unix, Linux Uptime and Reliability Increase: Patch Management Woes Plague Windows](#) Yankee Group survey finds IBM AIX Unix is highest in ...
- [Scalable, Fault-Tolerant NAS for Oracle - The Next Generation](#) For several years NAS has been evolving as a storage ...
- [Managing Software Intellectual Property in an Open Source World](#) This whitepaper draws on the experiences of the Black Duck ...
- [Open Source Security Myths Dispelled](#) Is it risky to trust mission-critical infrastructure to open source ...
- [Bringing IT Operations Management to Open Source & Beyond](#) Download this IDC analyst report to learn how open source ...

More ...

BREAKING NEWS

- [GNU tools touted for faster compiling target iPhone](#)
- [Palm Pre's price and launch date target iPhone](#)
- [Multi-core SoCs go superscalar](#)
- [Intel leaks details of "Pine Trail" Atom](#)
- [TI taps open source media streaming plugin](#)
- [Moblin v2 beta targets netbooks](#)
- [Linux-ready netbook touted as "Student rugged"](#)
- [Tutorial explores buildroot development](#)
- [Video overlay controller offers Linux-ready SDK](#)
- [Low-cost thin client offers choice of protocols](#)
- [Remote monitoring device offers 3G modem](#)
- [Mini-ITX mobo supports dual LVDS displays](#)
- [USB display technology heading for Linux](#)
- [Linux development service cracks DaVinci code](#)
- [Netbook boasts Linux multi-track recording suite](#)

More ...

Most popular stories -- past 90 days:

- [Linux boots in 2.97 seconds](#)
- [Tiniest Linux system, yet?](#)
- [Linux powers "cloud" gaming console](#)
- [Report: T-Mobile sells out first 1.5 million G1s](#)
- [E17 adapted to Linux devices, demo'd on Treo650](#)
- [Android debuts](#)
- [First ALP Linux smartphone?](#)
- [Cortex-A8 gaming handheld runs Linux](#)
- [Ubuntu announces ARM port](#)

DesktopLinux headlines:

- [USB display technology heading for Linux](#)
- [Game over for Linux netbooks?](#)
- [Linux Foundation relaunches Linux web site](#)
- [Dell spins lower-cost netbook](#)
- [Flashrom enables BIOS flashing via Linux](#)
- [Novell prepping Moblin version of SUSE](#)
- [Anti-malware software upgraded for Linux](#)
- [Mandriva's latest touted for fast boots](#)
- [\\$100 card converts old PCs into Linux thin clients](#)
- [Apple sued for squeezing wiki chat](#)

More ...

Also visit our sister site:

WindowsForDevices.com

Sign up for LinuxDevices.com's...

[XML](#)

[news feed](#)

Or, follow us on Twitter...



Related Stories:

- [SPECIAL REPORT: Linux kernel 2.6 arrives in embedded](#)
- [Creating embedded Linux filesystems with graphical tools](#)
- [Embedding Debian GNU/Linux in a 32MB CompactFlash](#)
- [How to boot Linux from an external USB or FireWire drive](#)

[\(Click here for further information\)](#)

FREE Whitepapers from WebBuyersGuide

8 Reasons to Solve Business Problems with Next-Gen Mobile Platforms

Enterprise customers today are poised to deploy mobile applications in greater numbers than ever before. As these businesses seek much more strategic deployments of mobile applications for improved worker productivity, enhanced customer service, and increased organizational efficiency, mobile enterprise platforms represent the core architecture behind such deployments.

FUEL Database on MontaVista Linux

Whether building a mobile handset, a car navigation system, a package tracking device, or a home entertainment console, developers need capable software systems, including an operating system, development tools, and supporting libraries, to gain maximum benefit from their hardware platform and to meet aggressive time-to-market goals.

Breaking New Ground: The Evolution of Linux Clustering

With a platform comprising a complete Linux distribution, enhanced for clustering, and tailored for HPC, Penguin Computing's Scyld Software provides the building blocks for organizations from enterprises to workgroups to deploy, manage, and maintain Linux clusters, regardless of their size.

Data Monitoring with NightStar LX

Unlike ordinary debuggers, NightStar LX doesn't leave you stranded in the dark. It's more than just a debugger, it's a whole suite of integrated diagnostic tools designed for time-critical Linux applications to reduce test time, increase productivity and lower costs. You can debug, monitor, analyze and tune with minimal intrusion, so you see real execution behavior. And that's positively illuminating.

Virtualizing Service Provider Networks with Vyatta

This paper highlights Vyatta's unique ability to virtualize networking functions using Vyatta's secure routing software in service provider environments.

High Availability Messaging Solution Using AXIGEN, Heartbeat and DRBD

This white paper discusses a high-availability messaging solution relying on the AXIGEN Mail Server, Heartbeat and DRBD. Solution architecture and implementation, as well as benefits of using AXIGEN for this setup are all presented in detail.

Understanding the Financial Benefits of Open Source

Will open source pay off? Open source is becoming standard within enterprises, often because of cost savings. Find out how much of a financial impact it can have on your organization. Get this methodology and calculator now, compliments of JBoss.

Embedded Hardware and OS Technology Empower PC-Based Platforms

The modern embedded computer is the jack of all trades appearing in many forms.

Data Management for Real-Time Distributed Systems

This paper provides an overview of the network-centric computing model, data distribution services, and distributed data management. It then describes how the SkyBoard integration and synchronization service, coupled with an implementation of the OMG's Data Distribution Service (DDS) standard, can be used to create an efficient data distribution, storage, and retrieval system.

[Home](#) | [News](#) | [Articles](#) | [Polls](#) | [Forum](#) | [About](#) | [Contact](#)



[Ziff Davis Enterprise Home](#) | [Contact Us](#) | [Advertise](#) | [Link to Us](#) | [Reprints](#) | [Magazine Subscriptions](#) | [Newsletters](#)
[Tech RSS Feeds](#) | [White Papers](#) | [ROI Calculators](#) | [Tech Podcasts](#) | [Tech Video](#) | [VARs](#) | [Channel News](#)

[Baseline](#) | [Careers](#) | [Channel Insider](#) | [CIO Insight](#) | [DesktopLinux](#) | [DeviceForge](#) | [DevSource](#) | [eSeminars](#) |
[eWEEK](#) | [Enterprise Network Security](#) | [LinuxDevices](#) | [Linux Watch](#) | [Microsoft Watch](#) | [Mid-market](#) | [Networking](#) | [PDF Zone](#) |
[Publish](#) | [Security IT Hub](#) | [Strategic Partner](#) | [Web Buyer's Guide](#) | [Windows for Devices](#)

[Developer Shed](#) | [Dev Shed](#) | [ASP Free](#) | [Dev Articles](#) | [Dev Hardware](#) | [SEO Chat](#) | [Tutorialized](#) | [Scripts](#) |
[Code Walkers](#) | [Web Hosters](#) | [Dev Mechanic](#) | [Dev Archives](#) | [Igrep](#)

Use of this site is governed by our [Terms of Service](#) and [Privacy Policy](#). Except where otherwise specified, the contents of this site are copyright © 1999-2009 [Ziff Davis Enterprise Holdings Inc.](#) All Rights Reserved. Reproduction in whole or in part in any form or medium without express written permission of Ziff Davis Enterprise is prohibited. Linux is a registered trademark of Linus Torvalds. All other marks are the property of their respective owners.