Red Hat Linux 5.1 vs. CentOS 5.1: ten years of change

D. Hugh Redelmeier
hugh@mimosa.com

Abstract

Red Hat Linux 5.1 was released in 1998. Almost ten years later, its direct descendant CentOS5.1 was released in 2007. How much has changed in the years since the first Ottawa Linux Symposium?

To investigate these changes, both systems were installed and used on the same hardware. What were the important changes? Did we use or abuse new resources as hardware developed along Moore's Law? Were the times as golden as some old-timers remember them to be? Can the youngsters still be taught a thing or two?

1 Introduction

In what ways has Linux changed? Most of us experience changes release by release. Taking a longer term view should yield a different set of insights.

Although recollection is a good tool, actual investigation seems worthwhile. To this end, I have installed Red Hat Linux 5.1 and CentOS5.1 on the same hardware. By using and examining these two platforms, I hope to investigate and compare them.

These platforms were chosen for several reasons. I have used each when they were current. Both were popular in their respective eras. One is a logical successor of the other (Red Hat Linux evolved to Red Hat Enterprise Linux, and CentOS is a clone of RHEL) so the codebases are strongly related. Finally, it is appealing that their version numbers happen to be identical.

2 Environment

Computer hardware made great capacity advances between the releases. Computers have become more pervasive in that same period. These environmental changes have affected the releases.

Date		1997 Oct	2007 June
price	$\div 5$	C$1965	C$400
brand		local shop	Acer Aspire E380
CPU	$\times 2$	AMD K6	AMD Athlon 64 x2
CPU clock	$\times 11$	200MHz	2200MHz
RAM	$\times 16$	64M	1024M
RAM type		PC66	PC2-5300
hard disk	$\times 39$	6.4G	250G
HD RPM	$\times 1.3$	5400	7200
optical		CD reader	DVD writer

This table illustrates the changes in hardware capacity. It sketches the dimensions of two computers that I bought to be Linux workstations. The first system's components were selected to be the most powerful I could get without leaving the mainstream whereas the second system was designed by Acer for normal home or office users.

The changes are large enough that they should drown out the effects of whether I selected a high-end or mainstream system at either time.

The changes in most dimensions are so large that one would expect them to be experienced as qualitative differences, not just quantitative. Think what it would be like if your house had forty times the floor space, the frequency of your piano's A key went up by a factor of eleven, or you desk had sixteen times the surface area.

The increased disk and RAM speeds are much less impressive. This suggests that algorithms, programs, and systems ought to be rebalanced to effectively use the new hardware.

In part, this paper was prompted by the question: how did Linux spend this increased capacity?

3 Installing RHL5.1 and CentOS5.1

To compare the distributions, I installed them both on the same computer.

The disparity in the hardware requirements made finding a computer that would support both a bit of a challenge. For example, only old video controllers are supported by RHL5.1; CentOS5.1 will only run on machines with perhaps 256M or more RAM (the graphical install requires 512M).

Just to see what would happen, I booted the RHL5.1 installation disk on my new HP Pavilion A6245n. It was quite confused by the 320G hard drive (fdisk, the kernel, and Disk Druid had varying wrong opinions of its size, based on various geometry lies) and about the 6G of RAM (it recognized only 64M). It saw one of the four CPU cores. Still, I expect RHL5.1 could have been installed.

For the actual installation, I chose a Compaq EN SFF box manufactured in 1999 April. I stuffed it with 320M of RAM and 120G of hard disk (it was probably originally shipped with 64M of RAM and a 6.4G hard drive). I expect that very few machines old enough to run RHL5.1 were initially assembled with enough RAM to install CentOS5.1.

The machine has no CD or DVD drive. Installation was through the network. In RHL5.1, the installation boot floppy can be told to find the installation tree via FTP, HTTP, or NFS. In CentOS5.1 the kernel has outgrown floppies so PXE netbooting was necessary for bootstrapping the installation.

RHL5.1 uses the LILO bootloader and this version does not use the extended int 13 features of modern BIOSes to access large disks. It could only access content on the first 1023 notional cylinders of the hard drive. So most of the drive was out of reach.

One approach to this problem is to create a separate `/boot` partition that is within the first 1023 cylinders. It appears as if RHL5.1 was not set up to support this. I did manage to accomplish this but there were a few odd failures that had to be dealt with. In the end, I used CentOS5.1's Grub to boot RHL5.1.

Even with the LILO problem dealt with, RHL5.1 seemed to only be able to use CHS mode to address the disk and thus was limited to the first 8.5G of the disk. After installation and updates, it seems to be able to use LBA addressing (thus supporting disks up to 137G). Making my way through a twisty maze of `fdisk` and `hdparm` seemed unrewarding so I did not resolve all of these mysteries.

RHL5.1 and CentOS5.1 cannot share swap partitions. RHL5.1 uses an older form of swapfile that is limited to 127M. From the standpoint of 2008, that limit is hard to believe.

To install CentOS5.1, I had to set up a PXE booting environment, something that I had never done before. This was made slightly more difficult by the fact that the documented technique for configuring CentOS5.1 as a boot server is to use the system-config-netboot package which turns out not to exist.

CentOS5.1 installed quite uneventfully, if slowly. The subsequent update process took an unreasonably long time. This seems to be a well-known problem even on current machines.

Lessons learned:

- It is possible to find hardware supported by distributions separated by a decade.
- Grub is a lot friendlier than LILO.
- The historical path of increasing disk size is littered with awkward limitations.

4 Experience with RHL5.1

In order to get current experience with RHL5.1, I attempted to use it to prepare this paper. This does not constitute a comprehensive survey but it was instructive.

Overall, I found using RHL5.1 was quite easy and effective. This depends on what the user is used to: someone habituated to current desktops would be much less comfortable. But even for me, the devil is in the details. What follows is a catalogue of issues.

RHL5.1 cannot be expected to support modern hardware. After all, the last changes to it were made in 1999 and they were just bug fixes. I used hardware from 1999 and found that worked.

The X desktop looks quite crude by current standards. It is based around FVWM. Looks don't matter very much. I didn't use the X desktop much, preferring to login from another desktop. That is mostly a reflection of the layout of my lab.

There is no SSH included in RHL5.1. I've grown very accustomed to its convenience and security so I missed it.

I tried to build a current version of OpenSSH on RHL5.1. I could have gone looking for a version of SSH's SSH (what I used in 1998) but I didn't really want to miss the years of bug fixes and other improvements.

I gave up on building OpenSSH because it demanded a newer version of Zlib and the addition of OpenSSL. It looked as if a cascade of backports would be required. This kind of barrier is probably typical when trying to backport current programs.

`rlogin(1)` worked. I hope that the security issues are not critical on my LAN. Unencrypted NFS is likely to be a juicier target.

JOVE is a text editor that I've used on UNIX-like systems for about 25 years. It has changed very little between the release of RHL5.1 and now. I built it on each system. CentOS5.1 was easy because the tarball includes a suitable `.spec` file for rpmbuild. For RHL5.1 a little work was required. The `.spec` file had comments that said how to change compile-time options to match RHL5.1 (mostly to do with POSIX conformance). One surprise was that RPM's macro processing seems to handle quoting differently—adjusting to that required an experimental approach.

The experience building JOVE would suggest that it isn't hard to make a program that can build in both environments. I don't think that this is accurate. JOVE had at least two advantages over most programs: it had been run on both systems before (albeit separated by many years), and its rate of change in that period has been very slow.

Building this paper using the OLS configuration did not work on RHL5.1. It failed with an unknown flag to `latex`: `-interaction=nonstopmode`. Even `xdvi` failed (missing fonts) on the `.dvi` file created by CentOS5.1. Being new to the LaTeX world, I decided not to attempt a work-around.[1]

The standard web browser is Netscape Communicator 4.08. Out of the box, the web pages I tried were blank or were missing a large part of their content (slashdot.org, google.ca). It turned out that turning off javascript helped considerably. The pages looked wrong or crude but the content was there. I had a look at some Gopher sites and they seemed fine. I would not like being limited to this browser these days.

In order to share files between the RHL5.1 and CentOS5.1 installations on the same machine, I tried to have each mount the others partition. CentOS5.1 could mount the RHL5.1 ext2 partition but RHL5.1 could not mount the CentOS5.1 ext3 partition, even though ext3, when properly unmounted, is supposed to be compatible with ext2. Mount's diagnostic was the infuriating "wrong fs type, bad option, bad superblock on `/dev/hda5` or too many mounted file systems." `dmesg(8)` showed the more specific `EXT2-FS:03:05: couldn't mount because of unsupported optional features`.

To solve the file sharing problem, I made a partition on another computer available via NFS. This worked well for both distributions.

5 Size of Programs

The two distributions share a lot of programs. How has their size changed?

I looked at all binary programs in `/bin`, `/usr/bin`, `/sbin`, and `/usr/sbin`. Symlinks were ignored but each hard link was counted. There were 1174 in RHL5.1 and 2413 in CentOS5.1. Of these, 655 were common to both (by name).

This attrition rate seems surprisingly high: 44% percent of the commands of RHL5.1 did not make it to CentOS5.1. A large number are probably explained by the fact that I did a "kitchen sink" installation of RHL5.1. Many of the programs that disappeared might have been short-lived marginal programs.

As reported by `size(1)`, the cumulative text space used by programs that were common to the two distributions has gone up by a factor of 2.7. Similarly, the size of data went up by 1.6 and BSS by 2.2.

Perhaps the programs found in `/bin` are in some sense more fundamental. Did they grow at a different rate? For programs found on both distributions and in `/bin` in either one of them, I find similar figures: a factor of 2.6 for text, 3.0 for data, and 1.4 for BSS.

I was surprised to find that for programs found in `/sbin` in either distribution, the growth was much

[1] *Ed. Note*: Workarounds would have failed due to requirements on a newer `geometry.sty` and other packages. —*Formatting Team*

higher: a factor of 5.3 for text, 2.6 for data, and 5.3 for BSS.

`bash(1)` is an important program, so it is worth looking at by itself. Text has grown by a factor of 2.1, data by 1.04, and BSS by 3.08. These figures are consistent with our cumulative ones.

I installed JOVE on both distributions. The text grew by a factor of 1.13; data and bss changed insignificantly. This was true whether the CentOS5.1 installation exploited the new POSIX capabilities or was configured identically to the RHL5.1 version.

This table shows programs whose text size shrank or grew by a factor larger than 10. `rmt` is included twice because RHL5.1 has two different versions.

Program	RHL	CentOS	Factor
gs	646943	3928	0.00607163
python	267795	2024	0.00755802
perl	419638	10186	0.0242733
symlinks	88660	6199	0.0699188
chroot	1377	14039	10.1954
tac	7802	82939	10.6305
repquota	5392	61268	11.3628
smbd	323322	4126046	12.7614
automount	14108	204301	14.4812
usleep	1495	22652	15.1518
mailstats	4113	63414	15.4179
warnquota	4347	70036	16.1113
smbpasswd	131262	2256936	17.1941
quotaon	3506	64400	18.3685
restore	53855	1040290	19.3165
praliases	2804	78458	27.9807
dump	33775	1101649	32.6173
rmt	4841	465708	96.2008
rmt	4296	465708	108.405
makedb	6136	815640	132.927

Each program that shrank did so because code moved to dynamic libraries and hence was not counted. In the case of `symlinks(8)`, the RHL5.1 version was statically linked for some reason.

Excluding these programs made only a modest change to the ratios: the text factor became 2.4, the data factor 1.7, and the BSS factor 2.1.

Almost every program uses libc. It has grown by a factor of two:

```
   text    data     bss
 591554   25728   48964
1282529   10072   11352
```

It seems as if there is a real expansion in the size of binaries but it is quite modest compared with the concurrent growth in hardware capacity.

As a point of comparison, I applied the same scripts to compare binary commands on CentOS5.1 i386 and x86_64. Of course there were many more commands in common. The cumulative text size went up by a factor of 1.24, the data size went up by a factor of 1.62, and the bss size went up by 1.07. I was surprised that the text of `/usb/bin/mbchk` was 139 times larger on x86_64. On the other hand `gedit` shrank by a factor of .40. In both cases the package versions were the same.

The RHL5.1 CD contains 528 packages taking up 298568 blocks. The CentOS5.1 DVD contains 2401 packages taking up 3570804 blocks. That is 4.5 times as many packages and 12 times as many blocks.

6 Functionality: the Qualitative Difference

Not only has hardware capability increased over the ten years, but open source developers have been working hard to exploit it. Here's a subjective list of important additions:

- Desktop integration, primarily GNOME and KDE. Or choose your own.
- Open Office
- support for a large portion of the proliferation of I/O devices and ways of connecting them (USB, FireWire, SATA, ...).
- scalable support for multiple processors
- scalable support for large memories and disk drives
- support for new architectures (although processor diversity on the desktop has gone down)
- complex and powerful tools for building internet services
- support for various media such as video (seriously constrained by patents). Official MP3 support has been dropped.

- Asterisk for telephony (not part of CentOS5.1, but available)
- MythTV for PVR replacement and much more (not part of CentOS5.1 but available)
- significant shift to higher level but less efficient languages such as Perl, Python, and Ruby.
- improved support for UNICODE (which itself has improved). But mention of support for Klingon has been dropped from the `unicode(7)` manpage.

7 Security or Don't try this at home

Best practices for security have changed quite a bit since RHL5.1 was released. ssh has replaced rlogin. Firewalls can be configured during installation. Servers are generally not installed listening to the internet. A system has evolved to publish security holes and patches for them in a timely fashion.

RHL5.1 did have `ipfwadm(8)` for implementing a firewall, but no canned configuration or easy-to-use configuration tool. Building a firewall out of this involved fairly arcane knowledge.

`rlogin(1)` was "kerberized" so its authentication was reasonable. It doesn't seem to pay attention to `~/.rhosts` by default. But the traffic is still passed in the clear.

Wietse Venema's TCP Wrappers is included and used.

There have been almost ten years of work discovering holes in the software without any patches for RHL5.1 (of course this isn't negligent: the intended fix is to move to a newer release). Maybe the holes are so obsolete that current attackers don't know of them or think of using them. But I would not count on that.

8 Bit Rot

My RHL5.1 disks were commercially pressed by Red Hat, Inc. They still work well. But when I went back in my archives to find the errata for RHL5.1, I found that a number of my burned CDRs had become damaged.

I stored the CDRs in paper envelopes with plastic windows. This was much more compact than storing them in jewel cases. Unfortunately the plastic widows deteriorated and began to stick to the label side of the disks. When I tried to remove the debris from a disk, it seemed as if the disk partially delaminated. I've put off attempted recovery until another day.

On the other hand, I have not been able to find `.iso` images for RHL5.1 on the internet. The errata are still available, but have been relocated. This is explained in a note on `http://www.redhat.com/security/updates/eol/`, at least for now.

It is unlikely that a store would stock a box of RHL5.1. Software does not seem to be like books in this regard. In fact, the legal regime for most commercial software makes used software stores legally suspect.

All this may seem inconsequential. But the problem is only going to get worse as time passes and yet there might be more interest in RHL5.1 in some years. This seems to be how antiques become so valuable: the objects must pass through a valley of interest during which most are lost, broken, or discarded. Mundane objects are the most subject to this attrition.

I touched on other types of bit-rot earlier: the difficulty in porting code back, dealing with the cumulative gradual (and not so gradual) changes to libraries and other requisites; the difficulty in running old operating systems on new hardware. At some point, genetic drift is sufficient for the systems to be considered different species.

The cure for bit rot is constant maintenance. It isn't clear who would find it worthwhile to perform this maintenance on RHL5.1.

9 The Structure of Growth

The subject of how systems grow is deep and interesting. You will have to look elsewhere for a thorough treatment. I commend Stewart Brand's book [3] on how buildings change as one place to look.

The skeleton of Linux is traditional UNIX [4], as elaborated by POSIX standards [2] [1]. Like a skeleton, these parts don't change very quickly. Most critically, little is removed from their interfaces since they are the bedrock on which other parts of the system are built.

This means that most programs from the RHL5.1 era should be easily moved to CentOS5.1.

One exception is that several programming language implementations have become more restrictive about what a proper program is. For example, many C programs need to be cleaned up to compile on current systems.

Some of the recent additions to Linux may turn out to be as important and (one hopes) long-lived and stable. HAL is an example. Various object models seem as important but not as convincingly right.

I expect that the vast majority of new packages in CentOS5.1 are not skeletal, and that is a good thing. That means they may well come and go without seriously disrupting other packages. This hypothesis should be investigated.

10 Observations

Since the code base for RHL5.1 has suffered serious bit rot, it seems unlikely that there are remaining practical applications for it.

The fact that RHL5.1 is strongly related to CentOS5.1, and yet so much smaller, suggests that it might be possible to subset the CentOS5.1 codebase to produce a modern lightweight system. A key advantage would be that the burden of maintaining the packages would be widely shared.

I would imagine that this approach would fit better with a project like Debian because it is directed by a diverse community of developers and not one coherent corporate strategy.

Understanding the trajectory from RHL5.1 to CentOS5.1 may help us prepare for the trajectory of the next ten years.

One thing that I have not observed is significant removal of complexity. It would be wonderful if that were possible, but it seems to violate some law of software thermodynamics. It seems to require starting over, and that seems too expensive for most situations.

References

[1] American National Standards Institute. *IEEE standard for information technology: Portable Operating System Interface (POSIX) : part 2, shell and utilities*. IEEE Computer Society Press, 1109 Spring Street, Suite 300, Silver Spring, MD 20910, USA, September 1993.

[2] American National Standards Institute. *IEEE standard for information technology: Portable Operating Sytem Interface (POSIX). Part 1, system application program interface (API) — amendment 1 — realtime extension [C language]*. IEEE Computer Society Press, 1109 Spring Street, Suite 300, Silver Spring, MD 20910, USA, 1994.

[3] Stewart Brand. *How Buildings Learn: What Happens After They're Built*. Penguin, 1995.

[4] Dennis W. Ritchie and Ken Thompson. The UNIX time-sharing system. *Communications of the Association for Computing Machinery*, 17(7):365–375, July 1974.

Proceedings of the Linux Symposium

Volume Two

July 23rd–26th, 2008
Ottawa, Ontario
Canada

Conference Organizers

Andrew J. Hutton, *Steamballoon, Inc., Linux Symposium, Thin Lines Mountaineering*
C. Craig Ross, *Linux Symposium*

Review Committee

Andrew J. Hutton, *Steamballoon, Inc., Linux Symposium, Thin Lines Mountaineering*
Dirk Hohndel, *Intel*
Gerrit Huizenga, *IBM*
Dave Jones, *Red Hat, Inc.*
Matthew Wilson, *rPath*
C. Craig Ross, *Linux Symposium*

Proceedings Formatting Team

John W. Lockhart, *Red Hat, Inc.*
Gurhan Ozen, *Red Hat, Inc.*
Eugene Teo, *Red Hat, Inc.*
Kyle McMartin, *Red Hat, Inc.*
Jake Edge, *LWN.net*
Robyn Bergeron
Dave Boutcher, *IBM*
Mats Wichmann, *Intel*