

Partitioning for Linux

Robert Heller
Deepwoods Software
Wendell, MA, USA *

March 17, 2006

Abstract

This article discusses some of the things to consider when planning the partitioning of a Linux system. These considerations include thoughts on how different directories will be used and how the usage of different directories might suggest how to partition your disk space. Also my thoughts on the relationship between backup strategies and partitioning are covered.

Contents

1 The unified UNIX directory tree.	1
2 One big partition vs. many smaller partitions.	2
3 Common partitioning schemes.	2
4 Additional thoughts about special sorts of server classes.	4
5 Conclusions.	4

1 The unified UNIX directory tree.

The overall file system structure on a UNIX system¹ is always a single tree, no matter how many or how few physical or logical disks might be connected to it. This differs from MS-Windows, where each disk (physical or logical) gets its own drive letter and has its own independent directory tree root. Under UNIX (or Linux), having more than one physical or logical disk is generally invisible to the user, since each disk is seamlessly integrated into the directory tree. This means there is never an absolute need to have everything on one disk

*Copyright (C) 2006 Robert heller.

¹Linux's file system structure is a UNIX file system structure.

and means it is possible to have any number of physical or logical disks and that having more than one physical or logical disk has no effect on how the operating system is laid out and has no effect in terms of how programs are installed and installed programs can always find their data and configuration files in standard places without needing to know which logical or physical disk a given directory resides on.

2 One big partition vs. many smaller partitions.

Many people would like to install Linux (or other UNIX style operating systems) all on a single partition. Yes, one can do this and generally it will work fine. There are, however, various reasons to install Linux/UNIX on a number of different partitions. These reasons relate to efficient backup strategies, user and system protection issues (protecting users from system problems and protecting the system from user problems and protecting users from each other) and to help the system cope with unusual disk usage issues by certain daemon processes.

3 Common partitioning schemes.

The most common partitioning scheme separates the directory tree into 4 classes: core system, system volatile, user static, and user volatile file areas. Generally this scheme starts out with four partitions²: root (mounted under /), var (mounted under /var), usr (mounted under /usr) and home (mounted under /home). Generally the root and var file systems can be relatively small (about 1 gig for a typical modern Linux distribution) and the usr and home file system can be somewhat larger (about 4-6 gig for usr and arbitrary large for home).

The root file system contains the core system files. This includes the kernel itself (generally in /boot), the base set of user and system utilities (in /bin and /sbin), the system configuration (in /etc), the essential libraries and kernel modules (in /lib), and the device files (in /dev). These directories contain the minimal set of files and directories needed to boot up a Linux system, at least into single user mode. Most of the files on this file system do not change much. Mostly only a few of the files under /etc change at all (generally manually by the system administrator), once the system has been installed and configured.

The var partition contains the system volatile data files. This includes daemon³ pid and lock files as well as the spool and data directories used by various daemons. This file system (mounted under /var) is expected to be constantly changing over the course of normal system operations, mostly by daemon processes.

²We will ignore the swap partition for the time being, since the swap partition is generally never part of the visible system directory tree.

³A UNIX server process is known as a "daemon".

The `usr` partition contains the bulk of the programs, libraries, documentation, and constant data used by the normal user(s) of the computer system. Except when new system software is installed or existing software is upgraded, files on this file system change not at all.

Finally, the home file system contains the personal files of the end-user(s) of the computer. The files on this partition are regularly being created, deleted, and changed, generally at the whim of the end-user(s).

So, the first pass at creating partitions for a Linux installation would be something like this subset of the file systems on my home desktop machine (36gig SCSI disk):

Filesystem	Size	Used	Avail	Use%	Mounted on
<code>/dev/sda1</code>	1011M	151M	809M	16%	<code>/</code>
<code>/dev/sda7</code>	5.9G	2.7G	2.9G	47%	<code>/home</code>
<code>/dev/sda3</code>	5.9G	3.5G	2.1G	63%	<code>/usr</code>
<code>/dev/sda5</code>	1011M	230M	730M	24%	<code>/var</code>

This uses roughly 14 gig worth of disk space. If you have additional disk space, you can make the `/home` file system larger⁴. I often limit the size of the `/home` partition, since that keeps my backups limited in size and use the "extra" space on the disk for a not backed up scratch file system. This scratch file system (which I usually mount under `/scratch`) contains random files I don't need to back up, such as files I downloaded from the Internet (which I can always download again) or things like image files from a photo CD (which are implicitly backed up on the CD itself).

Here is another example, from my laptop, which has a 40 gig IDE disk:

Filesystem	Size	Used	Avail	Use%	Mounted on
<code>/dev/hda1</code>	966M	285M	632M	32%	<code>/</code>
<code>/dev/hda5</code>	3.8G	255M	3.4G	7%	<code>/home</code>
<code>/dev/hda6</code>	3.8G	1.8G	1.8G	51%	<code>/mp3s</code>
<code>/dev/hda7</code>	24G	3.7G	20G	16%	<code>/scratch</code>
<code>/dev/hda3</code>	3.8G	2.2G	1.5G	61%	<code>/usr</code>

I've set aside a 4 gig partition to hold my MP3 files—I use my laptop as a sort of big bulky iPod. Virtually all of the MP3 files are ripped from CDs I own. Thus all of the files on the `/mp3s` file system are re-creatable, especially since I do have a backup of the scripts I used to extract and convert the audio files on the CDs on my desktop machine.

⁴But don't forget about a swap partition!

And yet another example from my model railroad control system:

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/hda1	966M	232M	685M	26%	/
/dev/hda6	7.6G	139M	7.1G	2%	/home
/dev/hda7	12G	3.1G	7.7G	29%	/scratch
/dev/hda3	5.7G	2.2G	3.2G	42%	/usr
/dev/hda5	1.9G	198M	1.6G	11%	/var
/dev/hda8	2.9G	1.8G	931M	67%	/WBL3

In addition to a (large) scratch file system, I also have a file system that contains ISO images of the operating system's distribution CDs (/WBL3–White Box Linux 3.0).

4 Additional thoughts about special sorts of server classes.

On a server machine, it is often reasonable and very desirable to set aside one or more file systems (generally mounted somewhere under /var) for the volatile data used by the server's daemons:

Daemon	Typical mount point	Description
Mail server (sendmail)	/var/spool/mail	Mail spool and queue files
	/var/spool/mqueue	
News server (inn)	/var/spool/news	News articles
Print server (lpd, cups)	/var/spool/lpd	Spooled print jobs
Web server (Apache)	/var/www	Web pages
Database server (PostgreSQL)	/var/lib/pgsql	Database files

5 Conclusions.

In conclusion, I'd like to say that there is no one "one size fits all" solution to how to partition your hard drive in preparation to the installation of Linux. There are many factors that need to be considered, including who will be using the system, how will they be using the system, how you plan on backing up files, how you (as system administrator) plan on resolving disk usage disputes amongst users, and considerations regarding server process disk needs. With a little careful thought and planning it is possible to come up with a well crafted partitioning scheme that will serve you for the life of the system (or at least the life of the hard drive(s)).