

A banner for TotalView 8.1. On the left, there is a blue button with the word "DOWNLOAD" in white, and a yellow box next to it that says "15-day free evaluation". In the center, the text reads "INTRODUCING TOTALVIEW 8.1 THE MOST POWERFUL MULTITHREAD DEBUGGER ON THE PLANET". On the right, there is the TotalView Technologies logo, which consists of three vertical bars of increasing height (blue, green, yellow) above the text "TOTALVIEW TECHNOLOGIES".

Since 1994: The Original Monthly Magazine of the Linux Community

[Home](#)

Linux Journal, January 2006: Build a Home Terabyte Backup System Using Linux

By [Duncan Napier](#) on Tue, 2005-11-29 02:00. [Software](#)

Build a low-cost, terabyte-sized backup server using Linux and back up your digital audio files, digital images and digital movie recordings.

A terabyte-plus backup and storage system is now an affordable option for Linux users. This article discusses options for building and configuring an inexpensive, expandable, Linux-based backup server.

Server Design

High-capacity disk drives are now widely available at prices that are incredibly cheap compared to those of only a few years ago. In addition, with so many Linux users now ripping CDs to disk, saving images from their digital cameras and recording video using digital camcorders and DVRs, such as MythTV, the need for backing up and archiving large amounts of data is becoming critical. Losing pictures and videos of your kids--or your audio music library--because of a disk crash would be a catastrophe. Fortunately, a high-capacity, Linux-based backup server can be built easily and cheaply using inexpensive disk drives and free software.

Virtually any home PC can meet the basic requirements for a backup server. If you have long backup windows or relatively small amounts of data, a slow computer is not an obstacle. Make sure your network is fast enough to transfer data within your backup window. For older equipment, the bottleneck for backups can be the disk data transfer bandwidth (30-150Mbps depending on disk technology).

Many consumer-level computers do not have cooling capacity for more than two internal hard disks. Most motherboards support a maximum of four onboard disks (often four ATA/IDE devices, but the two ATA/IDE and two SATA combination is becoming common). External USB high-capacity drives are also available. If your computer is older and has USB1, purchase an inexpensive USB2 PCI expansion card, which is ten times faster.

SCSI has fewer limitations, but it is expensive and has tended to lock purchasers in to "flavor-of-the-month" SCSI technologies. One option for disk expansion and upgrade is the Host Bus Adaptor (HBA), such as those made by Promise Technology. An HBA is a disk controller on a PCI expansion card. HBAs typically require no additional software, have their own BIOS and are not constrained by PC BIOS limits on disk size. HBAs let you put large disks (more than 120GB) into systems with legacy BIOSes, upgrade from ATA-33 to ATA-150 or mix ATA and SATA disks.

You may want to consider purchasing a dedicated fileserver. A bare-bones server capable of holding six disks (fully preassembled, no disks or OS) can cost less than \$1,500 US. With this initial investment, you can expand disk space as needed for less than \$0.80 per GB or grow by plugging in USB disks. Once you have decided how many disks you need, consider their space, cooling and noise requirements. Figure 1 shows an example of a backup system build from an old server. The system has well over a terabyte of storage capacity.



Figure 1. Storage array build from an old server (capacity of nine IDE disks, including five in a converted SCSI RAID stack). Additional IDE spots added with Promise HBA.

Even if you choose to build a server from scratch and populate it with high-capacity disks, you can expect costs for your terabyte-plus backup server still to be minimal in terms of its per-gigabyte price. This is because storage costs have decreased so dramatically. Table 1 provides a variety of different configurations for a backup server, along with estimated prices per gigabyte for each (note: prices are estimates and do not include taxes or shipping costs). As you can see from the table, costs for a new server equipped with more than two terabytes of storage can be built for a cost of less than \$1.50 per gigabyte. That will back up a lot of home movies, digital pictures and music files!

Table 1. Some Backup Options, with Estimated per-GB Costs

Type	Configuration	Capacity (TB)	Cost per GB (\$)
ATA/SATA Disk	Internal disk	0.4	0.56
Linux Desktop*	Three internal disks	1.2	0.84
Linux Desktop*	Three internal disks plus two USB external	2.0	0.73

LaCie 2TB Storage	Network server appliance	2	1.15
Linux Server**	Six internal disks	2.4	1.21
Linux Server**	Six internal plus two USB external	3.2	1.08

*Intel Celeron D 478 325 2.53GHz, 256MB of RAM. **Intel SC5275 chassis, Intel ATX Motherboard, dual-3GHz Xeon CPUs, 2GB of RAM.

Building the Server and Adding Additional Disk Drives

During the past few years, I have built backup servers using Red Hat Linux 9, but you can use any flavor of Linux. I use Red Hat 9 because it is stable, free, currently maintained (Fedora Legacy Project) and simple to install and configure. If you buy a new computer, you may have to use a more current version of Linux. I generally do not use RAID for low-budget systems where cost is paramount, but it is worth considering.

Software requirements for a Linux backup server are minimal. Basic network administration utilities (including the secure shell, SSH, and secure shell daemon, sshd) and rsync are required. rsync is a fast, incremental duplication/synchronization utility that comes with most Linux distributions. With SSH and rsync, you can carry out virtually all basic backup tasks. It is advantageous for a backup server also to be a fileserver, so I install Samba, the SMB fileserver as well. I use Samba because it is the default fileserver for MS Windows clients, and it also is readily accessible by any UNIX system (including Mac OS X) using a Samba client. If you have a homogenous UNIX network, you can use NFS, which I will not discuss here.

If you need to attach additional disks to your server, begin by making sure you have enough data (IDE/SATA/SCSI) cables and power lines to accommodate the expansion. Ensure that your drive is Linux-compatible (although most are). Turn off the power to your computer and disconnect the power cable. Physically attach the disk(s) to your computer. Linux should recognize the new disk(s) on boot. If your drive is not recognized, your disk is incompatible or you need to locate and install a driver for it. Check boot messages for new drives using the `dmesg` command. The boot message for an IDE drive may look like this:

```
hdb: ST3400832A, ATA DISK drive
```

All IDE/ATA (and some SATA) drives have the designation hdx, where the x is replaced with a letter of the alphabet (b in this case). Similarly, adding new USB or SCSI (and some SATA) disks gives boot messages indicating a new drive designation sdx, where the x is replaced by the appropriate letter.

Most Linux distributions come with a GUI disk manager. These disk managers let you define and format partitions (I generally use one partition per backup disk), assign mountpoints (for example, /data1, /data2) and mount the partition. The process also can be done from the command line using `fdisk` to create partitions.

Creating New Partitions

To create new partitions on hdb (above), type:

```
fdisk /dev/hdb
```

Type `m` at the `fdisk` prompt for a help summary. Typing `n` at the prompt asks about the new partition we are creating:

```
Command action
  e   extended
  p   primary partition (1-4)
p
```

For a single primary partition, type in `p`:

```
Partition number (1-4):1
```

You are then prompted for a partition number (type `1` for a single partition). Next, set the partition size by determining the first and last cylinder. Because we are using the whole disk, you should be able to select the default values (the first and last cylinders):

```
First cylinder (1-48641, default 1):
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-48641, default 48641):
Using default value 48641
```

Type `w` to write the partition table. You now have a partition, `/dev/hdb1`, that occupies the whole disk.

Next, format the partition in the filesystem of choice (mine is in the `ext3` format) using the `mkfs` command:

```
mkfs -t ext3 /dev/hdb1
```

Create a mountpoint for the new partition of your new disk (I'll call it `/data1`):

```
mkdir /data1
```

Mount the newly created ext3 partition:

```
mount -t ext3 /dev/hdb1 /data1
```

And, test reading and writing. Finally, add a line in /etc/fstab, the mount table, to mount automatically during the boot process:

```
# Device          mountpoint      fstype  options      freq  pass_no
/dev/hdb1         /data1          ext3    defaults     1     2
```

Software Configuration--rsync and SSH

rsync is included in most Linux distributions. You need rsync and SSH on both your backup client and server. Check to see whether rsync installed by typing `rsync` at the command prompt or check your list of installed packages. If you cannot find a binary distribution for your package, you can download the source code for rsync by following links on the rsync home page (see the on-line Resources).

The simplest way to run rsync over a network is as a standalone application using SSH for authentication. You can run rsync as a daemon with more features, but you won't need to in this case. I illustrate this here with a backup client named foo and a server named bar.

To replicate the directory /home on Linux machine foo with directory /data1/foo of backup server bar from client foo using rsync and SSH, type:

```
rsync -az /home -e ssh bob@bar:/data1/foo
```

You will be prompted for user bob's password, and then the foo /home directories are replicated to /data1/foo/home on bar (bob needs an account on the server and write permission for /data1/foo).

To avoid having to type bob's password each time, create a private/public key pair for SSH authentication without a password. This allows you to automate the login process.

Generating the Key Pair

On the machine you want to log in to (logged on as bob on bar in this case), type `ssh-keygen -d` to generate the key pair. Enter a password if the key will be accessible/readable to other users. Otherwise, press Return.

Change into the `.ssh` directory and copy the public key to the allowed list:

```
cd ~/.ssh
cp id_dsa.pub authorized_keys2
```

Copy the private key to the `.ssh` directory of the account on the machine you will be logging in from (for example, root user on foo). Remove the private key from bar (the machine you want to log on to):

```
scp id_dsa root@foo:~/.ssh/id_dsa
rm id_dsa
```

On the machine you're logging in from, start the SSH agent, and add the key to the agent's list (`ssh-add` asks for a password if you typed one in the first step above):

```
eval `ssh-agent`
ssh-add
```

You can now log in to account bob bar from foo without a password:

```
ssh bob@foo
```

You can run a script on foo to replicate foo on bar using bob's account on bar. You should read the documentation for `rsync`, which has numerous features (more than 70 command-line options). In particular, the `-delete` option can have disastrous consequences if misused. Listing 1 shows a seven-day incremental backup. Files altered or deleted on each day of the week are deposited in directories named for the day (set by `-backup-dir`). The most recent backup is stored in the directory `current`.

Listing 1. Full and Incremental `rsync`

```
#!/bin/sh
# This script does backups of foo to the backup server bar
# in a 7 day rotating incremental backup.
# Based on script by Andrew Tridgell
# directory to backup
BDIR=/home
# Remote directory on backup server
```

```

BACKUP_HOME=/data1/foo
# Backup login account on remote server
BACKUP_LOGIN=bob
# the name of the backup server
BSERVER=bar
BACKUPDIR=`date +%A`
OPTS="--force --ignore-errors --delete --backup
-backup-dir=$BACKUP_HOME/$BACKUPDIR -av"
export PATH=$PATH:/bin:/usr/bin:/usr/local/bin
# Dump output to backup file
date > /var/log/backup.$BACKUPDIR.log
# the following line clears the last week's incremental directory
[ -d /tmp/emptydir ] || mkdir /tmp/emptydir
rsync --delete -a /tmp/emptydir/
BACKUP_LOGIN@$BSERVER:$BACKUP_HOME/$BACKUPDIR/
rmdir /tmp/emptydir
# now the actual transfer
rsync $OPTS $BDIR BACKUP_LOGIN@$BSERVER:$BACKUP_HOME/current >>
/var/log/backup.$BACKUPDIR.log

```

If you prefer a compressed archive format, you still can run `tar` for a full backup over the network:

```
tar cvfz - /home | ssh bob@bar dd of=/data1/foo/current.tar.gz
```

and use the `-newer` option for an incremental tar backup.

`rsync` is more efficient than the `tar` command, because `rsync` copies only the differences between the current and previous copy of the data.

You can get by with `rsync` and `SSH` on most platforms (including MS Windows), but in reality, a fileserver setup is preferable, especially if you are running MS Windows clients. For MS Windows machines, a Windows backup application is preferable. The easiest way to do this is to run the backup to write to a share on the Samba server.

Software Configuration--Samba

If your Linux installation supports SMB file sharing, Samba is probably installed. If not, binaries are included with virtually all distributions. If this isn't the case with your distribution, or if you prefer to use the very latest Samba version, download the source code and compile and install. Official Samba distributions are available from the Samba home page (see Resources). Refer to the documentation there for installing and initially configuring Samba.

Once your backup server has Samba server installed, all Samba configurations are made by editing the `smb.conf` file, which is usually in `/etc/samba/smb.conf` or `/usr/local/samba/lib/smb.conf`. Graphical configuration utilities

like SWAT usually are included with Samba. See your documentation for information about starting or stopping Samba. You should configure your server to ensure that Samba starts when the server initially boots up.

Following our backup example above, on server bar, set up a simple smb.conf file or try appending the section below to the existing smb.conf file to define a share called bob:

```
[bob]
  comment = foo backup account
  path = /data1/foo
  valid users = bob
  public = no
  writable = yes
```

Next, add bob with any secure password as a Samba user (bob must have a Linux account as well as permission to read/write the /data1/foo directory):

```
smbpasswd -a bob
New SMB password: somepassword
Retype new SMB password: somepassword
Added user bob
```

For MS Windows clients, map the share \\bar\bob as a network drive in MS Windows using the user name bob and the SMB password for the bob Samba account. You then should be able to run backups to the mapped network drive. I typically use the free ntbakup software and set it up to write .bkf files to network storage. ntbakup comes free with Windows 2000 and XP and can run automated, regularly scheduled backups from the Windows client. Windows client-based backups have the advantage of backing up the entire state of the system (including the Windows registry).

You also can use Samba to serve files to most UNIX or Mac OS X clients. The smb client is installed by default in Mac OS X. In Linux distributions, make sure that the smb client package is installed. The smb share should be mounted onto the /backup mountpoint of machine foo:

```
mount -t smbfs -o username=bob,password=somepassword //bar/foo /backup
```

To have the backup drive mount when the system boots, place a line such as the following in /etc/fstab:

```
//bar/data1/foo          /backup          smbfs  rw,username=bob,
password=somepassword  0 0
```

Adding Off-Site Backup for Additional Security

To add an additional level of security, you may consider adding a second server to your overall backup plans consisting of a server that exists off-site, away from the home or office location where your primary backup server is located. This allows you to mirror your backup server to an off-site location once a week. That way, if you have a fire or some other catastrophe at your primary location, your data still will be available. Figure 2 shows a sample configuration for this setup.

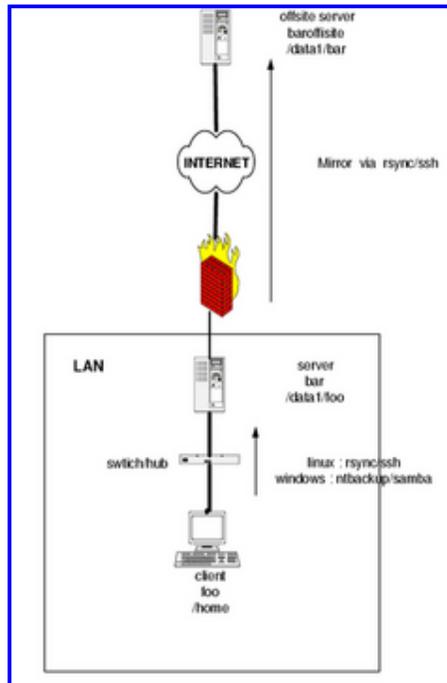


Figure 2. Example backup scenario with client foo, backup server bar and off-site mirror baroffsite.

Listing 2 is a basic script that mirrors the server bar with an off-site mirror baroffsite using rsync. Always set up backups to run automatically and on a regular schedule. Always keep logs of your backups, and always check the backup logs.

Listing 2. rsync Mirroring bar to baroffsite

```
#!/bin/sh
# Mirror /data1 on bar to /data1/bar on baroffsite.
#Backup directory on bar
BACKUP=/data1
#Backup directory on baroffsite
BACKUP_OFF=/data1/bar
```

```

# Give the day of week as name of backup
BACKUPNAME=`date +%A`
# Offsite server
BSERVER=baroffsite
# Backup account on backup server
BAC_ACC=backup
date > /var/log/backup.$BACKUPNAME.log
/usr/bin/rsync -avz --delete -e ssh $BACKUP $BAC_ACC@$BSERVER:$BACKUPOFF
>> /var/log/backup.$BACKUPNAME.log
# Email the log to administrator
cat /var/log/backup.$BACKUPNAME.log | mail -s 'Mirror Check'
backup_guy@mycompany.com

```

Monitoring Your Backup Server

In order to monitor your backup process and make sure your backups are running as scheduled (and that your backup server hasn't run out of disk space), it's important to put some automated monitoring and reporting into place. Listing 3 is a simple script that can be set up to run periodically via cron and send you a summary of the backups that have occurred and how much disk space is remaining on each of your partitions.

Listing 3. Simple Timestamp and Disk Space Lister

```

#!/bin/sh
# Check space on partitions
# List timestamps in chronological order
BACKUPS=/data1      #Identify directories to check
# Give the day of week as name of backup
BACKUPNAME=`date +%A`
#Timestamp
date > /var/log/backup.$BACKUPNAME.log
# Disk space on partitions
df -k > /var/log/backup.$BACKUPNAME.log
echo ' ' >> /var/log/backup.$BACKUPNAME.log
#List timestamps on backup server
# ls -lRt is much more verbose
ls -lt $BACKUPS/* >> /var/log/backup.$BACKUPNAME.log
# Email the log to administrator
cat /var/log/backup.$BACKUPNAME.log | mail -s 'Backup Check'
backup_guy@mycompany.com

```

Resources for this article: www.linuxjournal.com/article/8635.

Duncan Napier works as computer and instrumentation consultant in the Vancouver area of British Columbia.

» [add new comment](#) | [email this page](#) | [printer friendly version](#) | 26505 reads

Comment viewing options

Select your preferred way to display the comments and click "Save settings" to activate your changes.

[Excellent Write-Up](#)

Submitted by ncc74656m (not verified) on Fri, 2007-03-16 13:59.

For someone who is experienced in Linux, but has never done anything serious with it, I feel that this article is most beneficial and offers a lot of promise for the average newbie. It gives the user some sense of accomplishment when done without being complex and devastatingly hard.

I plan to do this for myself, and possibly a family member's small office when I have the chance and have proven the plausibility of it.

» [reply](#) | [email this page](#)

[Great, but what about backups?](#)

Submitted by systemloc (not verified) on Thu, 2007-02-08 07:45.

I've been running a 1TB usable RAID 5 box using linux and 6 200GB drives. Having the storage space is great, and it was very cheap to build, but here's the rub: How do you inexpensively build a backup solution for 1TB of changing data? I download, and I cull, and large amounts of data flows across that array, so how do I back up 200GB/month easily and cheaply? DVD-R is not an option, building another array is not an option IMO, since I want the option of keeping old backups. Blue-Ray or tape is what I'm inclined to look at, but both add \$500-600 to the cost off the top, plus media.

» [reply](#) | [email this page](#)

[ssh problem for not skilfull linux user](#)

Submitted by Anonymous (not verified) on Fri, 2006-11-10 08:02.

Hi All:

I found the article simple and easy to follow. But I had problems to implement it on my system. First of all I am not a skilfull linux user so I asked for help and found the solution.

Problem:

Skyscraper Ad

Subscribe Now!

Subscribe Now
<ul style="list-style-type: none">• Subscribe• New Archive CD• Renew• Customer Service

Magazine

- [Magazine Archives](#)
- [Write for Us](#)
- [Masthead](#)
- [FAQ](#)
- [Press Releases](#)

Advertise

- [Request Media Kit](#)
- [Editorial Calendar](#)
- [Online Advertising](#)

- [General Ad Info](#)

Resources

- [What Is Linux?](#)
- [Industry Events](#)
- [Other Resources](#)

Free eNewsletters

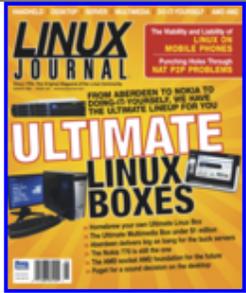
- [LJ Weekly Update](#)
- [Off the Shelf](#)

Popular content

Today's:

- [The Ultimate Distro](#)
- [Show Us the Code](#)
- [The Ultimate Linux/Windows System](#)

[\\$1.88 an issue!*](#)



[Start your risk-free subscription today!](#)

*based on a 2 yr U.S. sub

Navigation

- [by subject](#)
- [recent posts](#)
- [news aggregator](#)

In my machine and network environment, ssh authentication "as is" posted was not working.

Solution:

Check that you are using SSH 2. If your machine uses SSH 1 as default, as mine does, you should rewrite the line:

```
rsync -az /home -e ssh bob@bar:/data1/foo
```

with

```
rsync -az /home -e "ssh -2" bob@bar:/data1/foo
```

of course follow all the steps of key pair generation and so on.

hope it helps

» [reply](#) | [email this page](#)

[I don't know how to do...](#)

Submitted by paco69 (not verified) on Fri, 2006-07-28 03:13.

Hello,

I will want to adapt your tutorial to my needs.

My situation:

I have X server on linux system to backup. The backup is on a server windows solution.

How can i configure smb.conf? & What is the right way to write the line "mount -t smbfs"? & I am obliged to write in fstab? if yes, what do I have to write?

Thank you very much for your replies.

paco

ps:excuse me if my english is bad.

» [reply](#) | [email this page](#)

[Wow](#)

Submitted by [Ivan Minic](#) (not verified) on Sun, 2006-02-05 11:24.

The idea about this and realization of it is very nice indeed!



Coraid's
EtherDrive
Storage
less than
\$0.83/GB

[Click here to learn more](#)



Button 1 zone 6

REGISTER TODAY
@ java.sun.com/javaone

Text Ad Two - zone 10

[Intel Software Solutions](#)

» [reply](#) | [email this page](#)

[Report Script, change?](#)

Submitted by KGW (not verified) on Mon, 2006-01-16 02:28.

The line at the end of the report script assumes you have an MTA running on the local machine, I think. I don't and I guess that is why I get `./backup_report.sh: line 16: mail: command not found`

Is there a variation of this line that would send the generated report to my local mozilla thunderbird?

» [reply](#) | [email this page](#)

[problems with Generating the Key Pair](#)

Submitted by Ridgid (not verified) on Thu, 2006-01-12 00:45.

I tried the Generating the Key Pair section on my Fedora core 4 box connecting to a slackware 10.2 box. I had no luck getting the key pair to let me connect without a password. After some research I found this how-to on it that worked better for me.

<http://www.kernel-panic.org/wiki/SshKeygen>

Ridgid

» [reply](#) | [email this page](#)

[i had trouble with this..](#)

Submitted by [sergio_rrd](#) (not verified) on Thu, 2005-12-22 14:18.

i had some trouble with this..

i think the problem is:

```
BACKUP_LOGIN@$BSERVER:$BACKUP_HOME/$BACKUPDIR/
```

should be:

```
$BACKUP_LOGIN@$BSERVER:$BACKUP_HOME/$BACKUPDIR/
```

and

```
rsync $OPTS $BDIR BACKUP_LOGIN@$BSERVER:$BACKUP_HOME/current >>
```

[Development Products](#)
[Work right out-of-the-box](#)

[for Windows, Linux, Mac](#)

Text ad twelve - zone 22

[Find Bugs, Memory Leaks](#)

[TotalView is THE debugger for threaded or parallel code.](#)

[Get an evaluation copy.](#)

Text ad 11/z21

[Linux Journal for \\$1.88](#)

[Subscribe to LJ today for as low as \\$1.88 an issue.](#)

[linuxjournal.com/subscribe](#)

should be:

```
rsync $OPTS $BDIR $BACKUP_LOGIN@$SERVER:$BACKUP_HOME/current >>
```

» [reply](#) | [email this page](#)

[nice, how do you do restore though?](#)

Submitted by Anonymous (not verified) on Tue, 2005-12-13 12:52.

how do you do restore though?

» [reply](#) | [email this page](#)

[Another backuptool: http://www.bacula.org/](#)

Submitted by khoffrath (not verified) on Fri, 2005-12-02 16:22.

Bacula has a windows client which allows to backup data from windows clients without the need to set up a share (if i understand the docs correct).

Hope to find some time to test it out ;)

» [reply](#) | [email this page](#)

[Try KeyChain instead of passphraseless SSH keys](#)

Submitted by [Erik Postma](#) (not verified) on Thu, 2005-12-01 09:58.

Hi,

There is an alternative for passphraseless SSH keys that works quite well if you keep your linux-based backup client on for long times at a stretch: Keychain is a small program where you enter your SSH passphrase just once per power cycle. More info at <http://www.gentoo.org/proj/en/keychain/index.xml>

» [reply](#) | [email this page](#)

[Power Requirements](#)

Submitted by [timetrap](#) (not verified) on Thu, 2005-12-01 03:24.

I was just looking at starting the same project, but a modern (power saving) server will cost about 80 - \$100 a year for power (considering .07 - .10 a kWh). I think the linksys NSLU2 (which is already running linux) is a much

better option available for low power (1.5W instead of 60W). The only caveat; you need external USB drives to hook up to it.

Great article nonetheless, I hope that more people begin to build these so that the market place will have more (low power) options available.

» [reply](#) | [email this page](#)

[Tsync is also worth to mention](#)

Submitted by Anonymous (not verified) on Thu, 2005-12-01 02:48.

Basically is a modern rsync with many improvements. (Redundancy, peer to peer, ...). It's still in beta, but probably already more stable than rsync and other sync technologies (unison, ...).

» [reply](#) | [email this page](#)

[Diverse Disks](#)

Submitted by Chris (not verified) on Wed, 2005-11-30 15:43.

A comment on the purchase of HDD that I didn't see in the article.

For those of you considering building a home backup server, do not go to your nearest computer store and pick up 4 of the same type of drive.

As was pointed out in a previous comment, the failure rate of IDE drives increases after ~12 months.

If you purchase a number of HDD from the same manufacturer at the same time from the same location, you run the risk of getting a similar HDD failure on all your HDD at the same time due to possible manufacturing defects.

It is recommended to either purchase similar spec'd drives from different manufacturers or if you really want the same brand of drive, ensure that they drives you purchase are from separate manufacturing runs.

The diversity of drives will significantly reduce your chances of data loss in the event that a particular manufacturer has a manufacturing defect that causes the drive to fail.

Do some google lookups for IBM Deskstore or Fujitsu HDD failures in the past few years for examples.

Anyway, happy 'backuping' ;)

» [reply](#) | [email this page](#)

[Why Use Linux At All?](#)

Submitted by [Richard](#) (not verified) on Wed, 2005-11-30 15:35.

Why on Earth would I want to go to all this trouble when I can slap Server Elements NASLite (www.serverelements.com) onto a CD and 10 minutes later dispense with an operating system altogether?

» [reply](#) | [email this page](#)

[NASLite is Linux Powered](#)

Submitted by Kim Yamoto (not verified) on Thu, 2005-12-08 09:29.

NASlite is Linux based - 2.4.26 to be exact! I'll be impressed if someone can do the impossible and cram a Win32 app of that capability in 4M ramdisk. Let alone boot and run from a floppy disk.

» [reply](#) | [email this page](#)

[Re: Why Use Linux At All?](#)

Submitted by Michael Hearne (not verified) on Thu, 2005-12-08 00:42.

It's simple, here's the first clue; from the NASLite-in-a-Nutshell.pdf:

"At the DOS prompt, check to make sure that you can view the C drive."

This is a Linux group. We are Linux users. As a think-tank, we are creating our own solutions, and do not depend on commercial groups like the Microsoft Corp. to do it for us.

We share our sources and discoveries, and are not paid for it. However, our companies make more money for the effort, and as a result, we get raises.

The old Microsoft model worked like this: The programmer put a function into an exe file, and then sold the file.

The new Linux model works like this: The programmer shares his code and research; and as a result, his company profits, as do all concerned.

Can you produce the source code for this machine? I ask because there are many of us who also assemble our own hardware, and if we had the sources, then we would not have to purchase the machine at all.

The purpose of this article was to be a howto on assembling our own gear, and not to purchase Windows-based commercial stuff. I really don't understand the purpose of your question, unless it is an attempt to sell serverelement's (windows-based) equipment.

I think it's pointless here.

Michael Hearne

» [reply](#) | [email this page](#)

[You didn't read very close](#)

Submitted by Jim Budler (not verified) on Mon, 2005-12-12 14:59.

The NASLite in a Nutshell is an example of buying a settop box on e-bay for \$32 which has a built in flash disk. The DOSlike commands you listed were being used to load a Linux boot image into that flash disk. So \$32 for the machine, \$25 for the software, and whatever for the USB2 hard drive and you have an NAS. A dumb one but it will work.

It is not Windows based. The source code for the FOSS portion is available.

It isn't a solution I would use, as I would prefer to have the capabilities of having it be a member of a Domain instead of just a LAN community disk server, but I see no need to get so heated up about it.

You could emulate it easily using a live CD distro, and configure it for all that function, and have it able to be a member of a Domain. But that's work, and some people prefer a \$25 solution that "just works."

Jim

» [reply](#) | [email this page](#)

Exactly!

Submitted by Jeff (not verified) on Tue, 2005-12-13 09:55.

NASLite takes a minute to set up and just works. Best of all, it takes only 5 minutes to explain to a secretary how to administer it. Prior to using NASLite, RH was my choice, but when something goes wrong, I have to make a trip to correct the problem. With NASLite, a phone call usually resolves the issue.

Customer frustration is considerably lower this way.

NASLite just works.

» [reply](#) | [email this page](#)

Cheap Hardware and Low Time Investment

Submitted by Jeff (not verified) on Tue, 2005-12-06 10:46.

One can make a lot of arguments for and against Server Elements NASLite, but if you need to get a high capacity NAS server, built on low end hardware, there really are no alternatives. The damn thing runs in a mere 4M ramdisk. That exports your shares via CIFS, NFS, FTP and HTTP nicely and coherently.

I've been using it for 8 months on a 120MHz/64M Gateway box with 4x250G Maxtors. I had it set up and running in no time. Formatting the drives was the most time consuming task. Considering what I charge per

hour for my consulting services, the \$25 investment in the NASLite software was a no brainer. I've purchased multiple copies and installed it in many of my customer's locations. That way I get to do the job inexpensively for them and profitably for me.

Not a bad choice if you don't need a NAS fortress but a simple storage bin for a small office. I'd highly recommend it if you consider your time and customer's dollars at all valuable.

No affiliation with Server Elements, just like the product enough to speak up...

» [reply](#) | [email this page](#)

Why on Earth use NASLite

Submitted by [Patrick Rea](#) (not verified) on Sun, 2005-12-04 18:45.

Why would I pay for something like this when I can do it for free? \$25 and I can only run copy? I don't think so!

» [reply](#) | [email this page](#)

Doesn't NASLite use Linux

Submitted by Anonymous (not verified) on Thu, 2005-12-01 22:14.

Doesn't NASLite use Linux for OS portion?

» [reply](#) | [email this page](#)

RE: Why use Linux at all?

Submitted by Anonymous (not verified) on Thu, 2005-12-01 09:52.

Perhaps because

"By design, NASLite v1.x is a community file server and does not support features such as user management , the ability to join domains or disk quotas."

» [reply](#) | [email this page](#)

rdiff-backup

Submitted by [Anirban](#) (not verified) on Wed, 2005-11-30 09:13.

I am using LVM2, SAMBA, rdiff-backup and pyBackpack to setup my home backup system. It works quite well.

[Here](#) are the details.

» [reply](#) | [email this page](#)

When building a server with

Submitted by Juhani Tali (not verified) on Wed, 2005-11-30 09:09.

When building a server with many disks keep in mind, that hard drives use mostly 12V, but most psu-s have limited power on 12V. They give plenty of 5V or 3,3V, but not 12V. You should make certain, that the seek power consumption of all HDD is less than the psu@12V. Do not be fooled by "480W", it might not be enough for 8 hdd-s!

I highly recommend NOT to make a backup server on nonredundant IDE disks, especially if you use raid0. If you lose one disk you lose all the data! IDE hdd reliability starts to decline after ~12 months.

Also, consider the power consumption. 100+W of constant usage adds up.

» [reply](#) | [email this page](#)

failure rates for drives

Submitted by tgh (not verified) on Tue, 2005-12-06 11:36.

I've seen the "~12 months" show up twice now in this thread.

Most hard drives fail according in a "bathtub curve" pattern. Meaning, you will get a few that fail early in the lifecycle, then very few failures until the end of the lifecycle. The ones that failed early probably had manufacturing defects.

This is why you want to mix/match your drives in a RAID set from different manufacturing batches. So that a process glitch in the factory only affects one of the drives in the array. If all of your drives came from the same faulty factory line, they might all fail within the same timeframe. If this timeframe happens to be shorter than the recovery period for the RAID then you will lose everything on the RAID.

Using hot-spare drives shortens the recovery period (the RAID array can immediately start the rebuild as soon as a drive fails). That gives you good odds of getting redundancy again before a 2nd drive fails.

» [reply](#) | [email this page](#)

IDE disk statistics?

Submitted by [TimJowers](#) (not verified) on Tue, 2006-07-18 08:57.

Does IDE have spare sectors like SCSI? Does it allow reporting of disk problem statistics and remaining spare sectors?

Also, do IDE disks still not support simultaneous data requests or has a work-around been made? At the price point it seems IDE is the choice for data backups.

Thanks!

» [reply](#) | [email this page](#)

EVMS and BackupPC

Submitted by kbob (not verified) on Tue, 2005-11-29 21:11.

BackupPC is a preferable choice to rsync if (a) you are backing up Windows boxes, or (b) you want non-expert users to be able to restore their own files. BackupPC's HTTP interface is very nice for non-experts.

EVMS is probably preferable to LVM + RAID. It gives the same capabilities and even more configuration flexibility. I say "probably" because I built my backup server before EVMS was available.

» [reply](#) | [email this page](#)

BackupPC not so great on Windows

Submitted by Mick (not verified) on Thu, 2005-12-08 16:03.

Backup solutions like this work great on any other OS but Windows.

A list of limitations: <http://backuppc.sourceforge.net/faq/limitations.html>

These limitations are Windows limitations or similar and not really limitations of BackupPC.

I really recommend BackupPC is great, just keep these things in mind when you consider using it with Windows.

» [reply](#) | [email this page](#)

no it isnt

Submitted by Anonymous (not verified) on Wed, 2005-11-30 18:25.

backup pc will do linux too. not just windoze

» [reply](#) | [email this page](#)

[why not LVM or even raid](#)

Submitted by jeskritt (not verified) on Tue, 2005-11-29 17:50.

with so many disks in the machine you should use LVM or at least raid0 to make them look like one big uniform drive. Newer versions of RHEL or fedora give you the option to set these up if you manually partion with diskdruid.

With LVM you can even add disks to the LVM after the system is operational and increase the file system size. If you are worried about failing disks destroying all your data set up raid with hot spares or use raid5. As the article said, disks are cheap.

» [reply](#) | [email this page](#)

[Never use raid0. Ever.](#)

Submitted by Anonymous (not verified) on Wed, 2005-11-30 15:13.

Never use raid0. Ever.

» [reply](#) | [email this page](#)

[RAID 0 is very useful](#)

Submitted by Anonymous (not verified) on Fri, 2005-12-02 19:53.

If you are doing something where you want the fastest read write performance, with data that you can recover from another source if need be.

Let's say you DV video from your video camera to your RAID 0 partition. This video is still available from it's original source. You edit the video and then when you are done you compress the video and save it to a RAID 5 drive with a hot spare.

I am doing this with a set of five SCSI drives, all 18GB each. This gives me 90GB of the fastest available hard drive to play with. The read write speed is no longer a bottle neck. :D

» [reply](#) | [email this page](#)

[Not true.](#)

Submitted by Anonymous (not verified) on Thu, 2005-12-01 11:22.

RAID-0 is often used in combination with other RAID configurations (mainly when hardware RAID is involved) to gain high performance while minimizing risk. I've configured RAID "enclosures" with

multiple disk "trays" that are configured as RAID-5 w/hot-spare, with multiple trays spanning multiple HBAs, which have in turn been striped RAID-0. In order for a failure with data loss you would have to have 3 failures in one tray or more than 2 failures in multiple trays. Please don't say *never* when clearly the risks can be mitigated. RAID-5 does not replace proper backups either.

» [reply](#) | [email this page](#)

Better-er is rsync's --link-dest option

Submitted by Karl O. Pinc (not verified) on Tue, 2005-11-29 16:37.

You can dramatically improve the backup system using rsync's --link-dest option. Done right this will get you daily snapshots, like a full backup, but using only the disk space of an incremental backup.

(There are some issues with meta-information, file permissions and the like. mtree can be used to handle this.)

Rather than rolling your own, it could be better to use something like dirvish, which uses the --link-dest technique. <http://www.dirvish.org/>

» [reply](#) | [email this page](#)

A nice backup software

Submitted by Anonymous (not verified) on Tue, 2005-11-29 16:15.

A nice backup software backup to try is BackupPC. <http://backuppc.sf.net/>

» [reply](#) | [email this page](#)

Making this work with Win/Mac client installs

Submitted by Anonymous (not verified) on Fri, 2005-12-16 03:09.

Thank you for this. Always knew those storage boxes were obcenely priced. A question: Have you tried this with [Vembu StoreGrid](http://www.vembu.com) (see www.vembu.com) - it seems to support all OS and uses rsync/zlib etc with a friendly UI. How would this work with this Linux TB box - if I'm lazy and don't wanna go through implementing rsync etc manually? Esp since I'm planning a 'box' for backing up Win/Mac clients and another smaller FreeBSD server?

» [reply](#) | [email this page](#)

Go for it!

Submitted by Duncan_Napier (not verified) on Fri, 2005-12-16 11:09.

The setup presented is a completely generic one. I tried to show the most vanilla implementation of a terabyte-capacity storage system. The tools I presented are merely the most rudimentary utilities that come standard with virtually every Linux installation. If you have other tools that will run on Linux, then there is little to stop you implementing them as well. Have fun!

» [reply](#) | [email this page](#)

[Go for it!](#)

Submitted by Anonymous (not verified) on Tue, 2005-12-27 16:05.

Thank you for the article! As a Linux newbie, I learned a lot from this article. Although simplistic, I now have a backup server using linux.

Thanks again!

» [reply](#) | [email this page](#)



[Advertise](#) [Community](#) [Contact Us](#) [Privacy Statement](#) [Report](#) [Problems](#)

Copyright © 1994 - 2007 [Linux Journal](#). All rights reserved.