

Levels of Backup / DR / HyperSwap (Live Site) Recovery

Backup → Offline, must be restored onto storage/hardware to be usable

Disaster Recovery → Replication → Must be manually invoked, takes time, hardware must be available, but in a sense is "ready" for use: Sync and Async replication – Async is out of sync for a maximum period of time, referred to as an RPO or recovery point objective.

HyperSwap or High Availability (basically dual live site) → Involves replication that is already in use

In all cases, the needs of the business must be carefully examined to enable to most appropriate solution.

Everything has costs, the question is are the costs justifiable.

Backup, cloning, and Linux environments.

Linux does not have "system files" per say

Linux does not have a "registry" that can't be backed up easily.

The only parts of a linux system that might behave like this (ie open file issues, etc) are database files, such as mysql, postgresql, Oracle, DB2, etc

Almost all of a linux environment is just "files"

Because the linux environment is all "Files" it makes it easy to backup either locally, or remotely, or via a "cloud backup" provider.

If you **don't** backup - <http://adv-data.com/>

Cloud backup – the backup to save us all?

Great for off site

Horrible for full restoration

Great for multiple versions in most cases

Great for single file restores

Data in TB	Internet Speed in Mbps	Seconds	Minutes	Hours	Days
0.1	1.544	543303.6269	9055.0604	150.9177	6.2882
0.1	7	119837.2571	1997.2876	33.28813	1.387
0.1	15	55924.05333	932.06756	15.53446	0.6473
0.5	1.544	2716518.135	45275.302	754.5884	31.441
0.5	7	599186.2857	9986.4381	166.4406	6.935
0.5	15	279620.2667	4660.3378	77.6723	3.2363
0.5	100	41943.04	699.05067	11.65084	0.4855
1	1.544	5433036.269	90550.604	1509.177	62.882
1	7	1198372.571	19972.876	332.8813	13.87
1	15	559240.5333	9320.6756	155.3446	6.4727
1	100	83886.08	1398.1013	23.30169	0.9709

Examples:

Carbonite
BackBlaze
CrashPlan - <https://www.code42.com/store/>

Some offer "disk to door" services for "quick" restores, but quick is a relative term...

Tape backup:

Physical Tape

Density
Cost per GB
Capacity
Durable
Environmentally stable
Off-site is simpler than a hard disk drive
Not very economical on a small scale
Drive cost can be difficult to amortize over tape cartridge cost

Virtual Tape

Pretends to software that it's a tape drive
Uses "virtual" cartridges
In many cases uses deduplication technology to reduce storage needs
Can do virtual tape shipping to another location

Tape software

OpenSource – Bacula www.bacula.org
Commercial – IBM TSM - IBM Spectrum Protect
Commercial – Symantec BackupExec / NetBackup
Commercial – EMC DataDomain

Types of backups:

Full backup = Everything
~~Daily backup = Anything that changed on that day – Evil~~
Incremental backup = Anything that changed since the most **recent backup** (kinda
OK) Differential backup = Anything that changed since the most **recent full backup** (nice)

Remote sync to your own remote storage server

SSH

Outbound SSH from the "backup" server to the server being backed up.
Backup server can exist on dynamic IPs, and does not need to be remotely accessible.

RSYNC

Binary diff of files for transfer
Compute a checksum of the file on both sides, and see if they match
If they match, transfer for this file does not need to happen
If they don't match
Split the file logically into pieces

Run checksums on the pieces

Follow the steps again and again, and finally transfer only
the data you really need to transfer over the wire

Preserves permissions, links, etc very nicely

Very bandwidth friendly

Unix Hardlinks

Ability to point multiple "file names" at the same blocks on the disk

dirvish - <http://dirvish.org/>

hardlink tree to reduce storage usage

LAB idea one – use a dedicated drive on the Backup Server

"Debian Client"

In Vmware – Add a 50GB virtual drive, thin provisioned, single file

```
apt-get update && apt-get upgrade
```

```
apt-get install scsitolos ssh rsync dirvish parted
```

```
apt-get clean
```

you can use "rescan-scsi-bus" to rescan your drives without rebooting your VM

Partition the drive using parted with a **single partition with ext4 partition type**

We will setup the new drive mounted within /backup

```
parted -s -a optimal /dev/sdb mklabel msdos
```

```
parted -s -a optimal /dev/sdb mkpart primary 0% 100%
```

```
mkfs.ext4 /dev/sdb1
```

```
mkdir /backup
```

```
blkid /dev/sdb1 >> /etc/fstab, then edit your /etc/fstab to setup the mount
```

point for /backup based on the UUID of /dev/sdb1

Note UUID mounting is useful especially for external drives that may move

around a bit

```
mount -a
```

```
df -h
```

on your SERVER: adduser pima ← use a password you can remember or write

down

---- wait -----

```
wget http://server.cis225.a227.cis/cis225/dirv.tgz
```

```
tar xvfz dirv.tgz
```

```
chmod +x mydirvish.sh
```

```
./mydirvish.sh
```

```
dirvish --vault server --init ← before you can do
```

this command, you need to have done the commands on the server below

```
ls -la /backup/server ← you should see a "date named"
```

folder

```
dirvish --vault server ← to do a second backup
```