

# ○ ● ● Backing up in Style

Fast and efficient backups with rdiff-backup

Presented to the Fraser Valley Linux Users  
Group

March 11, 2004

By Alan Bailward <[alan@ufies.org](mailto:alan@ufies.org)>

# ○ ● ● Why Backup?

- Preserve data
- Save work and time
- Reinstalling is annoying
- Gives you a warm fuzzy feeling

# ○ ● ● No One Backs Up

“Backups occur immediately following a system crash”

– unknown



- Backing up is a pain to set up
- Many users and sysadmins just hope nothing happens to their system
- I know this first hand
- ... but restoring is worse



# It's Not the Backup

... it's the restore

Everyone can backup (in theory), but the restore is where people fall down.

# ○ ● ● Restoring

---

- Something that might never be needed
- Hard to test
- Scary to simulate

# ○ ● ● Backup Methods

- A hacked together shell or perl script involving tar and scp that grew from a one liner
- Commercial tool such as Arkira...
- Alternative Open Source solution

# ○ ● ● What's Needed?

- Scriptable
- Debuggable
- No more disk space used than needed
- Backup to any media/destination
- Incremental backups help here

# ○ ● ● Incra-what-al?

Incremental backups are backing up *only* changed information

## ● Saves

○ disk space

○ bandwidth

○ time

## ● But...

○ harder to restore

○ can't just copy files back

# ● ● ● Enter rdiff-backup

- <http://rdiff-backup.stanford.edu>
- Backs up one directory to another, local or remote
- Open Source, compiles under POSIX Unix operating systems (and cygwin)

# ○ ● ● Technologies

---

- Written in python
- Mainly a combination of two technologies
  - (r)diff
  - rsync

# ○ ● ● What is (r)diff?

- Creates a “patch” file with only the changes from one version to another
  - Diff used mostly for code (linux kernel)
  - Less information needed to go from one version to another
- rdiff does binary diffs/patches

# ○ ● ● What is rsync?

- Mirrors files and attributes
- Local or remote
- Tunnel over ssh/rsh/sockets
- Very fast algorithm
- Sends only differences

# ● ● ● Rsync Syntax

- `rsync [OPTIONS] SRC DEST`

- `rsync [OPTIONS] SRC [USER@]HOST:DEST`

- `rsync [OPTIONS]] [USER@]HOST:SRC DEST`

- And all combinations...

- For example

- `rsync -avz --exclude '*~' /home/ ajb@server:/backup/`

# ○ ● ● Putting it Together

- rdiff-backup uses...
  - rsync for fast mirroring
  - rdiff for text/binary increments

# ○ ● ● Features

---

- Creates a mirror
- Keeps increments
- Preserves all information
- Space/bandwidth efficient
- Transparent data format

# Usage

## Rdiff-backup Syntax

```
rdiff-backup [OPTIONS] SRC DEST
```

## Example of local to local...

```
rdiff-backup /foo /bar
```

## Local to remote...

```
rdiff-backup /home ajb@host::/backups
```

## Remote to local...

```
rdiff-backup ajb@host::/home /backups
```

## With more options...

```
rdiff-backup --include /usr/local --exclude /usr /  
ajb@host::/backup
```

# ○ ● ● Increments 1

## ● Normal incremental backup example

- Sunday - Full backup
- Monday - increment 1
- ...
- Saturday - increment 6



- Pain in the butt to restore if you crash on Saturday
- Hard to get to latest data

# ○ ● ● Increments 2

- Rdiff-backup

- stores a mirror of current files

- older changes stored as reverse diffs via rdiff

- even binary files

# ● ● ● Restoring with RD-B

● As easy as copying from the backup dir: `cp /back/file /foo/file`

● Or `scp ajb@host:/back/file /foo/file`

● Using `rdiff-backups` functions:

```
rdiff-backup --restore-as-of now host.net::/  
remote-dir/file local-dir/file
```

● Or by date: `rdiff-backup -r 10D host.net::/  
remote-dir/file local-dir/file`

# ○ ● ● Other Functionality

- `-list-increments`
- `-print-statistics`
- `-remove-older-than`
- The *Man* page is your friend

# ○ ● ● “Niggely Bits”

---

- include/exclude syntax not totally intuitive
- Need dev series for `--remove-older-than nB` syntax
- `.12.6` incompatible with `.13.x`
- Python, ick! :-P



# My Script

```
#!/bin/sh

# remote host details
REMOTE_HOST=arcterex.net
BACKUPDIR=/var/backups/data
PGBACKUP=$BACKUPDIR/pg.sql
MYBACKUP=$BACKUPDIR/mysql.sql

echo "Dumping fdisk output..."
sfdisk -l > $BACKUPDIR/sfdisk-output.txt

# dump the postgres database
echo "Dumping Postgres database..."
PGUSER=postgres PGPASSWORD=password su postgres -c \
    "/usr/bin/ pg_dumpall -d > $PGBACKUP 2> /dev/null"

# dump the mysql database
echo "Dumping Mysqldatabase..."
mysqldump -u root -ppassword --all-databases -c > $MYBACKUP

# directories to backup
MAININCLUDES="--include $BACKUPDIR \
    --include /root \
    --include /home/alan \
    --include /var/www \
    --include /etc \
    --include /var/mail "

#optional directories to backup with local not remote
EXTRAINCLUDES="
    --include /home/silverstr \
    --include /home/raskal "
```

```
# cont...

echo "Executing local rdiff-backup..."
rdiff-backup --print-statistics -v1 $MAININCLUDES $EXTRAINCLUDES --exclude / / \
    /var/backups/ufies.org
# catch the return code for error checking later
RET1=$?

echo "Executing remote rdiff-backup..."
rdiff-backup --print-statistics -v1 $MAININCLUDES --exclude / / \
    alan@${REMOTE_HOST}:/mnt/share/var/backups/ufies.org
# catch this return code too. 0 means ok, non-zero means not ok
RET2=$?

echo "Removing old backups..."
# if everything went well then remove old backups
if [ "$RET1" = 0 ] ; then
    echo "Local backup completed, removing older than 1 month"
    rdiff-backup --print-statistics -v1 --remove-older-than 1M --force \
        /var/backups/ufies.org
else
    echo "Bad return code for local backup"
fi
if [ "$RET2" = 0 ] ; then
    echo "Remote backup completed, removing older than 7 days"
    rdiff-backup --print-statistics -v1 --remove-older-than 7D --force \
        alan@${REMOTE_HOST}:/mnt/share/var/backups/ufies.org
fi
```

# Resources

- <http://rdiff-backup.stanford.edu>
- `man rdiff-backup`
- <http://samba.anu.edu.au/rsync/>



Questions?