BorgBackup

Mohammad Fazeli

May 19, 2021

Frame Title

"I found the Holy Grail of backups."
(Stavros K. about Attic-Backup, 8/2013)

Backup Desiderata

- Take very little space
- Encrypted
  - Backup place hacked means everything is gone
- Multiple versions of data
- Pruning versions By Grandfather-Father-Son
- Deduplicated, reduced space

Backup Desiderata: Not using Borg

- Disk failure shouldn't cause data loss
- Server failure shouldn't effect availability
- Disks have bit-rot

My Solution

Borg + BTRFS RAID1 on at least two locations BTRFS RAID1
removes bit-rot problem two locations solves availability

What is BorgBackup

BorgBackup (short: Borg) deduplicating backup
program(compressed and encrypted authenticated)

Borg Features

- cli backup tool, separate GUIs
- good arch / platform / fs support
- deduplication
- compression
- auth. encryption
- read from locally mounted fs
- store to local fs or to remote borg
- Fuse-mount backup archives
- HW-accelerated crypto

Borg as a Project

- community project
- FOSS (BSD license)
- Python 3.7(current, 3.6-9 supproted)
- for speed: a little Cython & C
- good docs (for users, devs)
- good test coverage(83% today), travis CI
- github: borgbackup/community
- Pays money for security bugs(10,500$ paid today)

In Practice

- size of postgres db
- 48 hourly
- 20 daily
- Weekly 6
- Monthly ...
- 78 instances each 2GB
- all:153.06 GB
- compressed 41.24 GB
- deduplicated 22.50 GB

In Practice

- media folder of a big django project
- 3 instances each about 63 GB
- 182.70 GB compressed all instances(incompressible mostly already compressed images)
- 60.13 GB deduplicated across time

In Practice

- a little changing postgres db
- 12 instances
- 7 daily
- 4 weekly
- 1 monthly
- 2.75 GB all
- compressed 180MB
- deduplicated 30 MB

In Practice

- a wordpress site files
- 32 instances
- 80 GB total
- 70 GB compressed(incompressible mostly images)
- deduplicated 2.68 GB

In Practice

- a wordpress site db
- 32 instances
- 5.5 GB
- 420 MB compressed
- 204 MB deduped

In Practice

- a lot of mail boxes
- 20 instances
- 20 daily
- total 727GB
- 348 GB compressed(compressible since it is text)
- deduped 17.5 GB

In Practice

Total size: 1145 GB
Fitted in 103GB

It scales well

one of developers of borg borg info
ssh://borg@myserver/repos/myrepo
Original size 22.76 TB
Compressed size 18.22 TB
Dedup size 486.20 GB
Unique chunks 6305006
Total chunks 272643223

Where to put

Own server with ssh and free space:
install borg, configure, done!
No own remote server? No problem:

- ▶ rsync.net (ssh, cli)
- ▶ hetzner's storage box
- ▶ borgbase.com (ssh, web, easy)
- ▶ local repo + rclone to cloud

Deduplicating

dedup does **NOT** depend on:

- ▶ file/directory names:
  - ▶ move file even between machines, no change required
- ▶ Whole file hash and timestamps:
  - ▶ only a small part of a file changed,only the change, VMs , or raw disks, only changes are synced.
- ▶ Place of chunk in a file:
  - ▶ stuff moved around inside a file , doesn't increase it.

Encryption

Enc :

- ▶ Tampering/Corruption detection by HMAC or blake2b
- ▶ 256 bit AES- CTR
- ▶ uses OpenSSL(libcrypto)
- ▶ Passphrase and a key, could be separate
- ▶ PDKDF2 100K rounds

Compression

- none
- lz4: fast, low compression(faster than none :-|)
- zstd: from high speed low compression to slow and high compression
- zlib: medium speed and compression
- lzma: low speed, high compression

Safe

borg uses:
- checksums
- transactions
- fs: syncing, atomic ops
- backup repo- log-like KV store
- checkpoint while backing up

```
# initialize a repository:
borg init /tmp/borg

# create a "first" archive inside this repo (verbos
borg create --progress --stats\

 /tmp/borg::first ~/Desktop


# create a "second" archive (less verbose):
borg create /tmp/borg::second ~/Desktop

# even more verbose:
borg create -v --stats /tmp/borg::third ~/Desktop
```

```
# list repo / archive contents:

borg list /tmp/borg
borg list /tmp/borg::first

# extract ("restore") from an archive to cwd:

mkdir test ; cd test
borg extract /tmp/borg::third

# simulate extraction (good test):
borg extract -v --dry-run /tmp/borg::third

# check consistency of repo:
borg check /tmp/borg
```

language

```
		# list repo / archive contents:
		
		borg list /tmp/borg
		borg list /tmp/borg::first
		
		# extract ("restore") from an archive to cwd:
		
		mkdir test ; cd test
		borg extract /tmp/borg::third
		
		# simulate extraction (good test):
		borg extract -v --dry-run /tmp/borg::third
		
		# check consistency of repo:
		borg check /tmp/borg
	
```