restic Encrypted backups

START HERE

`restic init`
Create a repository

`restic backup <path>`
Take a snapshot

`restic snapshots`
List snapshots

`restic restore latest --target <dir>`
Restore newest

`restic forget --keep-daily 7 --prune`
Apply retention

`restic check`
Verify repository

`restic mount <dir>`
Browse as files

`restic unlock`
Clear stale locks

Files your data → `backup` → **Snapshot** deduplicated, encrypted → `forget` → **Unreferenced data** still in repo → `prune` → **Space freed** repo shrinks

CONCEPTS

Set RESTIC_REPOSITORY and RESTIC_PASSWORD_FILE once, then every command works without -r.

Every snapshot is a full view, but identical chunks are stored once, so repeated backups are small and fast.

Lose the password and the data is gone: there is no recovery. Store it separately from the repo.

REPOSITORY

`restic init`
Create a new repository

`restic -r <repo> <command>`
Select a repository for one command

`export RESTIC_REPOSITORY=<repo>`
Set the default repository

`export RESTIC_PASSWORD_FILE=<file>`
Read the password from a file

`--password-command "<cmd>"`
Get the password from a command

`restic -r sftp:<user>@<host>:<path> init`
Repository over SFTP

`restic -r s3:https://<host>/<bucket> init`
S3-compatible repository
Needs AWS_ACCESS_KEY_ID and AWS_SECRET_ACCESS_KEY in the environment.

`restic -r rest:https://<host>:8000/ init`
Repository on a REST server

`restic -r rclone:<remote>:<path> init`
Any backend via rclone

`restic key list`
List repository passwords

`restic key add`
Add another password

BACKUP

`restic backup <path>...`
Back up paths into a new snapshot

`restic backup -n -v <path>`
Dry run, show what would be added

BACKUP (cont.)

`restic backup --exclude "<pattern>" <path>`
Skip matching files

`restic backup --exclude-file <file> <path>`
Read exclude patterns from a file

`restic backup --tag <tag> <path>`
Label the snapshot

`restic backup --one-file-system <path>`
Stay on one filesystem

`restic backup --exclude-caches <path>`
Skip dirs marked CACHEDIR.TAG

`restic backup --exclude-larger-than 1G <path>`
Skip big files

`restic backup --files-from <file>`
Read the path list from a file

`restic backup --skip-if-unchanged <path>`
No new snapshot if nothing changed

`<cmd> | restic backup --stdin --stdin-filename <name>`
Back up piped data
A failing producer can still yield a snapshot; --stdin-from-command avoids this.

SNAPSHOTS

`restic snapshots`
List snapshots

`restic snapshots --tag <tag>`
Filter by tag

`restic snapshots --host <host> --path <path>`
Filter by host and path

`restic ls latest`
List files in a snapshot

`restic ls -l latest <dir>`
Long listing of one directory

`restic find <pattern>`
Search all snapshots for a file

`restic diff <id1> <id2>`
Compare two snapshots

`restic stats`
Show restore size

`restic stats --mode raw-data`
Show actual repo size

`restic tag --add <tag> <id>`
Add a tag to a snapshot

SNAPSHOTS (cont.)

`restic forget <id>`
Remove one snapshot

RESTORE

`restic restore <id> --target <dir>`
Restore a snapshot

`restic restore latest --target <dir>`
Restore the newest snapshot

`restic restore latest --host <host> --path <path> --target <dir>`
Newest for a host and path
--path only picks the snapshot, it does not limit what is restored.

`restic restore latest:<subdir> --target <dir>`
Restore only one subfolder

`restic restore <id> --include "<path>" --target <dir>`
Restore only matching files

`restic restore <id> --exclude "<path>" --target <dir>`
Restore all but matching files

`restic restore <id> --dry-run -vv --target <dir>`
Preview a restore

`restic dump latest <file>`
Print one file to stdout

`restic mount <dir>`
Mount the repo via FUSE
Linux needs fuse and fusermount; the mountpoint must be outside the repository.

`fusermount -u <dir>`
Unmount (or press Ctrl+C)

RETENTION

`restic forget --keep-last <n>`
Keep the n newest snapshots

`restic forget --keep-daily <n> --keep-weekly <n> --keep-monthly <n>`
Tiered retention policy

`restic forget --keep-yearly <n>`
Keep one per year

`restic forget --keep-within 30d`
Keep everything of recent days

`restic forget --keep-tag <tag>`
Never remove tagged snapshots

RETENTION (cont.)

`restic forget --dry-run <keep-flags>`
Preview what would be removed

`restic forget --prune <keep-flags>`
Forget and reclaim space

`restic forget --host <host> --tag <tag> <keep-flags>`
Limit policy to a subset

`restic prune`
Delete unreferenced data

`restic prune --dry-run`
Preview prune

MAINTENANCE

`restic check`
Verify repo structure

`restic check --read-data`
Read and verify all pack data
Slow and downloads everything on remote backends.

`restic check --read-data-subset 10%`
Verify a random share of data

`restic unlock`
Remove stale locks

`restic copy --from-repo <src> <id>`
Copy snapshots to another repo

`restic version`
Show version

OPTIONS AND ENVIRONMENT

`--json`
Machine-readable output

`-q / -v`
Quieter or more verbose output

`RESTIC_PASSWORD`
Password in the environment
Visible to other processes of the same user; prefer a password file.

`AWS_ACCESS_KEY_ID / AWS_SECRET_ACCESS_KEY`
S3 credentials

RECIPES

Nightly backup script

```
export RESTIC_REPOSITORY=
sftp:backup@host:/srv/restic
export RESTIC_PASSWORD_FILE=/
etc/restic/pw
restic backup /home /etc --
exclude-caches --tag nightly
restic forget --keep-daily 7 --
keep-weekly 4 \
  --keep-monthly 12 --prune
```