restic Encrypted backups

CONCEPTS

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

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

> 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

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

`restic -r s3:https://<host>/<bucket> init`
S3-compatible repository

BACKUP

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

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

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

SNAPSHOTS

`restic snapshots`
List snapshots

`restic ls latest`
List files in a snapshot

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

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

RESTORE

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

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

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

`restic mount <dir>`
Mount the repo via FUSE

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

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

`restic prune`
Delete unreferenced data

MAINTENANCE

`restic check`
Verify repo structure

`restic unlock`
Remove stale locks