



Linux Ubuntu 26.04 Manual Pages on command 'restic-forget.1'

\$ man restic-forget.1

restic backup(1) restic backup(1)

NAME

restic-forget - Remove snapshots from the repository

SYNOPSIS

restic forget [flags] [snapshot ID] [...]

DESCRIPTION

The "forget" command removes snapshots according to a policy. All snapshots are first divided into groups according to "--group-by", and after that the policy specified by the "--keep-" options is applied to each group individually. If there are not enough snapshots to keep one for each duration related "--keep-{within-}" option, the oldest snapshot in the group is kept additionally.

Please note that this command really only deletes the snapshot object in the repository, which is a reference to data stored there. In order to remove the unreferenced data after "forget" was run successfully, see the "prune" command.

Please also read the documentation for "forget" to learn about some important security considerations.

EXIT STATUS

Exit status is 0 if the command was successful. Exit status is 1 if there was any error.

Exit status is 10 if the repository does not exist. Exit status is 11 if the repository is already locked. Exit status is 12 if the password is incorrect.

OPTIONS

-l, --keep-last=*n* keep the last *n* snapshots (use 'unlimited' to keep all snapshots)

-H, --keep-hourly=*n* keep the last *n* hourly snapshots (use 'unlimited' to keep all

hourly snapshots)

-d, --keep-daily=0 keep the last n daily snapshots (use 'unlimited' to keep all daily snapshots)

-w, --keep-weekly=0 keep the last n weekly snapshots (use 'unlimited' to keep all weekly snapshots)

-m, --keep-monthly=0 keep the last n monthly snapshots (use 'unlimited' to keep all monthly snapshots)

-y, --keep-yearly=0 keep the last n yearly snapshots (use 'unlimited' to keep all yearly snapshots)

--keep-within= keep snapshots that are newer than duration (eg. 1y5m7d2h) relative to the latest snapshot

--keep-within-hourly= keep hourly snapshots that are newer than duration (eg. 1y5m7d2h) relative to the latest snapshot

--keep-within-daily= keep daily snapshots that are newer than duration (eg. 1y5m7d2h) relative to the latest snapshot

--keep-within-weekly= keep weekly snapshots that are newer than duration (eg. 1y5m7d2h) relative to the latest snapshot

--keep-within-monthly= keep monthly snapshots that are newer than duration (eg. 1y5m7d2h) relative to the latest snapshot

--keep-within-yearly= keep yearly snapshots that are newer than duration (eg. 1y5m7d2h) relative to the latest snapshot

--keep-tag=[] keep snapshots with this taglist (can be specified multiple times)

--unsafe-allow-remove-all[=false] allow deleting all snapshots of a snapshot group

--host=[] only consider snapshots for this host (can be specified multiple times) (default: \$RESTIC_HOST)

--tag=[] only consider snapshots including tag[,tag,...] (can be specified multiple times)

--path=[] only consider snapshots including this (absolute) path (can be specified multiple times, snapshots must include all specified paths)

-c, --compact[=false] use compact output format

-g, --group-by=host,paths group snapshots by host, paths and/or tags, separated by comma (disable grouping with "")

-n, --dry-run[=false] do not delete anything, just print what would be done

--prune[=false] automatically run the 'prune' command if snapshots have been removed

--max-unused="5%" tolerate given limit of unused data (absolute value in bytes with suffixes k/K, m/M, g/G, t/T, a value in % or the word 'unlimited')

--max-repack-size="" stop after repacking this much data in total (allowed suffixes for size: k/K, m/M, g/G, t/T)

--repack-cacheable-only[=false] only repack packs which are cacheable

--repack-small[=false] repack pack files below 80% of target pack size

--repack-uncompressed[=false] repack all uncompressed data

--repack-smaller-than="" pack below-limit packfiles (allowed suffixes: k/K, m/M)

-h, --help[=false] help for forget

OPTIONS INHERITED FROM PARENT COMMANDS

--cacert=[] file to load root certificates from (default: use system certificates or \$RESTIC_CACERT)

--cache-dir="" set the cache directory. (default: use system default cache directory)

--cleanup-cache[=false] auto remove old cache directories

--compression=auto compression mode (only available for repository format version 2), one of (auto|off|max) (default: \$RESTIC_COMPRESSION)

--http-user-agent="" set a http user agent for outgoing http requests

--insecure-no-password[=false] use an empty password for the repository, must be passed to every restic command (insecure)

--insecure-tls[=false] skip TLS certificate verification when connecting to the repository (insecure)

--json[=false] set output mode to JSON for commands that support it

--key-hint="" key ID of key to try decrypting first (default: \$RESTIC_KEY_HINT)

--limit-download=0 limits downloads to a maximum rate in KiB/s. (default: unlimited)

--limit-upload=0 limits uploads to a maximum rate in KiB/s. (default: unlimited)

--no-cache[=false] do not use a local cache

--no-extra-verify[=false] skip additional verification of data before upload (see documentation)

--no-lock[=false] do not lock the repository, this allows some operations on read-only repositories

-o, --option=[] set extended option (key=value, can be specified multiple times)

--pack-size=0 set target pack size in MiB, created pack files may be larger (default:

\$RESTIC_PACK_SIZE)

--password-command="" shell command to obtain the repository password from (default:

\$RESTIC_PASSWORD_COMMAND)

-p, --password-file="" file to read the repository password from (default:

\$RESTIC_PASSWORD_FILE)

-q, --quiet[=false] do not output comprehensive progress report

-r, --repo="" repository to backup to or restore from (default: \$RESTIC_REPOSITORY)

--repository-file="" file to read the repository location from (default: \$RESTIC_REPOSITORY_FILE)

--retry-lock=0s retry to lock the repository if it is already locked, takes a value like 5m or 2h (default: no retries)

--stuck-request-timeout=5m0s duration after which to retry stuck requests

--tls-client-cert="" path to a file containing PEM encoded TLS client certificate and private key (default: \$RESTIC_TLS_CLIENT_CERT)

-v, --verbose[=0] be verbose (specify multiple times or a level using --verbose=n`, max level/times is 2)

SEE ALSO

restic(1)

generated by restic generate

Jan 2017

restic backup(1)