Certbot Let's Encrypt client

CONCEPTS

> Certbot needs root. Point your server at /etc/letsencrypt/live/<name>/, never at the archive copies.

Request certonly or run
↓ challenge
Validate HTTP-01 or DNS-01
↓ issue
Store live/<name>/
↓ renew
Renew timer + hooks

OBTAIN

`certbot certonly -d <domain>`
Get a cert, do not install it

`certbot --nginx -d <domain>`
Get a cert and edit nginx config

`-d <a> -d <b>`
Several names on one cert

`--cert-name <name>`
Name the cert (file paths, housekeeping)

`--expand -d <old> -d <new>`
Add a name to an existing cert

`--staging`
Use the test CA (no rate-limit risk)

AUTHENTICATORS

`--webroot -w <dir>`
Write challenge files into a live web root

`--standalone`
Run a temporary server on port 80

`--dns-cloudflare / --dns-route53`
DNS-01 via a DNS provider plugin

INSPECT AND RENEW

`certbot certificates`
List certs, names, expiry and paths

`certbot renew`
Renew every cert that is due

INSPECT AND RENEW (cont.)

`certbot renew --dry-run`
Test renewal against the staging CA

`certbot renew --cert-name <name>`
Renew one cert only

`--force-renewal`
Renew even if not due

`--deploy-hook <cmd>`
Run after a successful renewal only

MANAGE CERTS

`certbot delete --cert-name <name>`
Delete a cert and its renewal config

`certbot revoke --cert-name <name>`
Revoke a cert at the CA

FILES AND HOOKS

Under /etc/letsencrypt

Path	Holds
`live/<name>/fullchain.pem`	Cert plus chain: use this
`live/<name>/privkey.pem`	Private key
`live/<name>/cert.pem, chain.pem`	Parts, rarely needed
`archive/<name>/`	Numbered history
`renewal/<name>.conf`	Saved options for renew
`renewal-hooks/{pre,deploy,post}/`	Executable scripts
`cli.ini`	Global defaults

> Hook scripts in renewal-hooks/ run in alphabetical order. Deploy hooks run only when a cert was actually renewed.

> Most packages install a systemd timer (certbot.timer) or cron job; check before adding your own.