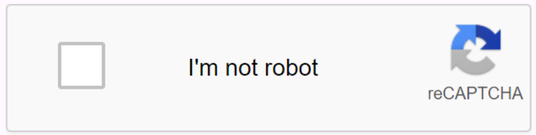

Continue

Certbot certonly manual renew

Home About Certbot FAQ Documentation Support Source Code Donate eff ≡ Other Versions v: Home About Certbot FAQ Documentation Support Source Code Donate EFF ≡ Other Versions v: Use: Certbot [SUBCOMMAND] [Options] [-d DOMAIN] [-d DOMAIN] ... Certbot can obtain and install HTTPS/TLS/SSL certificates. By default, it will try to use the webserver to both obtain and install certa. The most common subCOMMUNADS and flags are: acquire, install and renew certificates: (default) run Get and install cert in your current certonly webserver Get or restore cert, but do not install it restore Restore all previously obtained certs that are close to expiration -d DOMAINS List of domains separated from comma to obtain certa for - apache Use Apache plugin to authenticate and install - Standalone Run Standalone Authentication Webserver - nginx Use the Nginx Authentication and Installation Plugin -- webroot Place files in the webroot authentication server folder -- manually Get certs interactively or using a shell script hook -n Run inaucipherively -- test-cert Get cert test from setup server -- dry-run Test renew or certonly without saving certs to disk management certificates : Certificates View certe information you have from Certbot recall Revoke certificate (supply - cert-put) Delete Delete certificate manage your account using Code : Register Let's Encrypt ACME account - agree to agree to the ACME server subscriber contract -m email address for important account notifications optional arguments: -h, --help show this help message and exit -c CONFIG_FILE, --config CONFIG_FILE path to config file (default: /etc/letsencrypt/cli.ini and ~/.config/letsencrypt/cli.ini) -v, --verbose This flag can be used multiple times to gradually increase output verbosity, e.g. (default: -2) -n, --uninteractive, --uninteractive running without searching for user input. This may require additional command line flags; the client will try to explain which ones are needed if it finds one missing (default: False) - force interactive Force Certbot to be interactive, even if it detects that it is not working on the terminal. This flag cannot be used with a renewal subcommand. (default: False) -d DOMAIN, --domains DOMAIN, --domain DOMAIN Domain names to apply. For multiple domains, you can use multiple -d flags or enter a comma-separated list of domains as a parameter. (default: Ask) --certname login certificate name. Only one certificate name can be used per Certbot ride. To see the certificate names, run certbot certificates. When creating a new certificate, the name of the new certificate is specified. (default: None) -- dry-run Perform a test drive of the client, obtaining a test of (invalid) certs, but does not spare them on the disk. It can currently only be used with certonly and renew subcommunicables. Keep in place: Although tries to avoid any permanent changes to the system, it is not Side Effect Free: If used with webserver authenticator plugins such as apache and nginx, it does so and then returns temporary configuration changes to get a certs test, and reload webservers for implementation, and then restore those changes. It also calls -- pre-hook and -- post-hook commands if defined because they may be needed to accurately simulate renovations. --hip renewal commands are not called. (Default: False) -- Debug Challenges After setting up a challenge, wait for the user to enter before submitting CA (default: False) -- preferred challenges PREF_CHALLS Sorted, comma list of preferred challenges to use during authorization with the most desirable challenge first listed (e.g. Not all plugs support all challenges. See details in . ACME challenges are inami, but if you choose http and not http-01, Certbot will automatically select the latest version. (default: []) --user-agent USER_AGENT set up a custom set of user agents for the client. A number of user agents allow CA to collect high-level statistics on success rates by the OS and add-on. If you want to hide the server OS version from the Let's code server, set this to . (default: CertbotACMEClient/ 0.13.0 (Ubuntu 16.04.2 LTS) Authenticator / XXX Installer / YYY) automation: Arguments for automation of execution and other tweaks - keep-until-expiring, --keep, --reinstall If the requested cert matches the existing cert, always keep the existing cert until the restoration occurs (for the 'run' subcommand, this means reinstaling the existing cert). (Default: Ask) -- Expand If an existing cert covers a subset of required names, always expand it and replace it with additional names. (Default: Ask) -- the version shows the version number of the program and the output -- forced renewal -- renewal by default If the certificate already exists for the required domains, restore it now, regardless of whether it is close to expiration. (Often -- keep-until-expiring is more appropriate). Also implies --expand. (Default: False) -- Renew with new domains If the certificate already exists for the required certificate name but does not match the required domains, renew it now, regardless of whether it is close to expiration. (default: False) -- Allow a subset of names When performing domain validation, do not consider it a failure if authorizations cannot be obtained for a strict subset of required domains. This can be useful for enabling multi-domain renewals to succeed even if some domains no longer point to this system. This option cannot be used with - csr. (default: False) -- agree to an agreement from ACME subscribers (default: Ask) -- duplicate Allow the creation of certificate breaks that duplicate existing (both can be restored in parallel) (default: Fake) -- only os-packages (certbot-auto only) install OS package dependencies, and then stop (default: False) -- (certbot-auto only) prevent certbot-car script. Script. themselves on the newly released versions (default: Upgrade automatic) --no-bootstrap (certbot-auto only) prevent the certbot-auto script from installing OS-level dependencies (default: Install OS-level dependencies immediately, but exit if the user says 'No') -q, --silence Silence all come out except errors. Useful for cron automation. It implies -inselithing. (default: false) security: Security parameters and server settings -- rsa-key size N RsA key size. (default: 2048) -- Must-staple Adds OCSP Must Staple Extension to Certificate. Autoconfigures OCSP Stapling for supported settings (Apache version >= 2.3.3). (Default: Incorrect) -- Redirect Automatically redirect all HTTP traffic to HTTPS for newly requested vhost. (Default: Ask) -- Do not automatically redirect all HTTP traffic to HTTPS for newly requested vhost without redirection. (default: Ask) -- hsts Add a stern traffic safety header to each HTTP response. Force your browser to always use SSL for your domain. He's defending himself against SSL stripping. (default: False) -- uir Add a Content-Security-Policy header: upgrade-insecure-requests to each HTTP response. Force your browser to use https:// for each http:// resource. (default: None) --staple-ocsp enables OCSP Stapling. A valid OCSP response is merged into a certificate that the server offers during TLS. (default: None) -- strict permissions require that all configuration files are owned by the current user; Only if your configuration is unsafe somewhere such as /tmp/ (default: False) testing: The following flags are only intended for testing and integration. --test-cert, --staging Use server staging to obtain or recall test (invalid) certs; equivalent -server (default: False) -- debug Show tracebacks in case of errors and allow certbot-automatic execution on experimental platforms (default: False) -- non-verify-ssl Disable ACME server certificate verification. (default: False) --tls-sni-01-port TLS_SNI_01_PORT Port that is used during tls-sni-01 challenges. It only affects the port certbot listens to. The compliant ACME server will still try to connect to port 443. (default: 443) -- http-01-port HTTP01_PORT Port used in the http-01 challenge. It only affects the port certbot listens to. The compliant ACME server will still try to connect to port 80. (default: 80) --break-my-certs Be ready to replace or restore valid certs with invalid (testing/setting) certs (default: False) paths: Arguments that change execution paths and servers - cert-path CERT_PATH Path to where cert is saved (with auth - csr), installed from or revoked. (default: None) -- key-path KEY_PATH Path to private key for cert installation or revocation (if account key is missing) -- fullchain-path FULLCHAIN_PATH Trailing path to full certificate chain plus chain). (default: None) -- The chain path is CHAIN_PATH the path to the certificate chain. (default: None) --config-dir --config-dir Configuration directory. (default: /etc/letsencrypt) -- work-dir WORK_DIR Working directory. (default: /var/lib/letsencrypt) --logs-dir LOGS_DIR Logs directory. (default: /var/log/letsencrypt) --server SERVER ACME Directory Resource URI. (default: Management: Various subumunds and flags are available for certificate management: Certbot-managed Certificate List delete All certificate renewal files Renew all certificates (or those listed with --cert-name) revoke certificate specified with - cert-path update_symlinks Recreate symlinks in your /etc/letsencrypt/live/directory run: Options for getting and installing certs certonly: Options for modifying how to obtain a certificate -- CSR CSR Path to Certificate Signing Request (CSR) in DER or PEM format. Currently - CSR works only with a 'certonly' subcommend. (default: No) renew: The renewal subcommend will try to renew all certificates (or more precisely, certificate heralds) that you previously received if they are close to expiration and print a summary of the results. By default, restore will reuse the options used to generate profits or recently successfully restore each erotica of the certificate. You can try it with a '-dry-run' first. For more fine-grained controls, you can restore individual lineages with a certonly subcommend. Hooks are available to run commands before and after renewal; see for more information about them. --pre-hook PRE_HOOK to be guided in the shell before receiving any confirmations. Intended primarily for restoration, where it can be used to temporarily turn off a webserver that may conflict with a standalone plug-in. This will only be called if the certificate is actually to be obtained/renewed. When renewing several certificates that have identical pre-hooks, only the first one will be performed. (default: None) -- a post-hook POST_HOOK command to run in the shell after trying to obtain/renew the certificate. It can be used to implement restored certificates or to restart any server that has been stopped - pre-hook. This is only done if an attempt has been made to obtain/renew the certificate. If multiple restored certificates have identical post-hooks, only one will run. (default: None) -- renew-hook RENEW_HOOK command that will once run in shell for each successfully restored certificate. For this command, the shell variable $RENEWED_LINEAGE will indicate the configuration of a live sub-survey containing new certs and keys; the $RENEWED_DOMAINS variable will contain a list of restored space-delimited cert domains (default: None) -- disabling hip validation Usually commands specified for -- pre-hook /--post-hook/--renew-hook validation, to see if the programs being run in $PATH, so that errors can be caught early, even when the hooks are not yet working. Validation is simplified and fails if you use advanced shells, so you can use this switch to disable it. (default: False) certificates: Certbot-managed certificate list deletes: Options for deleting certificates revoke: Options for cert revocation -- reason {keycompromise, affiliation changed, superseded,indefinite, termination offoperation} Specify a reason for revoking the certificate. (default: 0) register: Options for registering and modifying your account -- register unsafely without email Stating this flag allows you to register an account without an email address. This is strongly discouraged, as in the event of a crucial loss or compromise on your account, you will irrevocably lose access to your account. You will also not be able to receive notification of impending expiration or withdrawal of certificates. Updates to the Subscribers Agreement will continue to affect you and will take effect 14 days after the website updates are published. (default: false) -- Registering registry verb updates indicates that details related to an existing registration, such as an email address, should be updated instead of registering a new account. (default: Fake) -m E-mail, --e-mail E-mail used for registration and recovery contact. (default: Ask) -- email Share your email address with EFF (default: None) -- non-eff-email Don't share your email address with the EFF (default: None) neregister: Options for deactivating your account. --account ACCOUNT_ID account ID to use (default: None) install: Options for modifying how certificates are deployed config_changes: Options for the control that changes are displayed -- num NUM How many past revisions do you want to display (default: None) restore: Options for restoring server configuration changes -- N Revert configuration checkpoints N number of checkpoints. (default: 1) add-ons: Options for the plug-in subcommittee -- init Initialize plugins. (default: Fake) -- prepare initialization and prepare add-ons. (Default: False) --Limit authentication only to authentication add-ons. (default: None) -- Only limit installers to installer plug-ins. (default: None) update_symlinks: Recreates cert and key symlinks in /etc/letsencrypt/live, if you changed them manually or edited update configuration plugins: Select add-ons: Certbot client supports extensive plug-in architecture. See certbot plugins for a list of all installed plugins and their names. You can force a specific add-in by setting options below. Run - Help will list flags specific to this <plugin_name> add-on. - Configurator CONFIGURATOR The name of the plug-in, which is both an authenticator and an installer. It should not be used in a --authenticator or -- installer. (default: Ask) -AUTHENTICATOR authentication plug-in name. (default: None) -i INSTALLER, --installer INSTALLER Installer plug-in name (also used to locate domains). (default: None) -- apache Get and install Using Apača (default: Fake) -- nginx Get and install certs using Nginx </plugin_name> </plugin_name> Fake) -- Get certs on your own using a standalone webserver. (default: Fake) -- manual Provide strenuous manual instructions for obtaining certa (default: false) -- webroot Get certs by placing files in the web directory uprooted. (default: False) nginx: Nginx Web Server plugin - Alpha -- nginx-server root NGINX_SERVER_ROOT Nginx server root directory. (default: /etc/nginx) --nginx-ctl NGINX_CTL Path to the 'nginx' binary, which is used to trust and retrieve the nginx version number. (default: nginx) standalone: Spin the temporary webserver manual: Authentication via manual configuration or custom shell scripts. When using shell scripts, you must provide an authentication script. The environment variables available to this script are $CERTBOT_DOMAIN that contains the verifying domain, $CERTBOT_VALIDATION that is a valid string, and $CERTBOT_TOKEN which is the file name of the resource requested when running the HTTP-01 challenge. An additional cleaning script can also be provided and can use an additional variable $CERTBOT_AUTH_OUTPUT containing stdout output from the auth script. --manual-auth-hook MANUAL_AUTH_HOOK Path or execution command for authentication script (default: No) -- manual-cleanup-hook MANUAL_CLEANUP_HOOK Path or cleaning script execution command (default: None) -- manual-public-ip-logging-ok Automatically allows public IP logging (default: Ask) webroot: Place files in webroot directory -- webroot-path WEBROOT_PATH, -w WEBROOT_PATH public_html / webroot put. This can be determined multiple times to handle different domains; each domain will have a webroot path that preceded it. For example: '-w /var/www/example -d example.com -d www.example.com -w /var/www/thing -d thing.net -d m.thing.net' (default: Ask) -- webroot-folder WEBROOT_MAP JSON dictionary domain mapping on webroot paths; this implies -d for each entry. You may have to escape from your shell. For example: --webroot-folder '{eg1.is,m.eg1.is:/www/eg1/, npr2.is:/www/eg2}' This option is merged with, but has an advantage over, -w/ -d entries. Currently, if you put a webroot-map in a configuration file, it must be on a single line, such as: webroot-folder = {example.com:/var/www}. (default: {}) apache: Apache Web Server plugin - Beta -- apache-enmod APACHE_ENMOD Path to the Apache 'a2enmod' binary. (default: a2enmod) -- apache-dismod APACHE_DISMOD road to Apache 'a2dismod' binary. (default: a2dismod) --apache-le-vhost-ext APACHE_LE_VHOST_EXT SSL vhost configuration extension. (default: -le-ssl.conf) --apache-server-root APACHE_SERVER_ROOT server root directory. (default: /etc/apache2) --apache-vhost-root APACHE_VHOST_ROOT Apache server VirtualHost configuration root (default: /etc/apache2/sites-available) -- apache-logs-root APACHE_LOGS_ROOT Apache server logs directory (default: /var/log/apache2) --apache-challenge-location APACHE_CHALLENGE_LOCATION Directory path for challenge configuration. (default: APACHE_HANDLE_MODULES Installer is working on enabling the necessary modules for you. (Currently only Ubuntu/Debian) (default: True) -- Apache-handle-sites APACHE_HANDLE_SITES Let the installer handle you with the sites they allow. (Currently only Ubuntu/Debian) (default: Exact) null: Null Installer Installer