
certbot-dns-cloudflare Documentation

Release 0

Certbot Project

Dec 06, 2018

Contents:

1	Named Arguments	3
2	Credentials	5
3	Examples	7
4	API Documentation	9
4.1	certbot_dns_cloudflare.dns_cloudflare	9
5	Indices and tables	13
	Python Module Index	15

The *dns_cloudflare* plugin automates the process of completing a `dns-01` challenge (DNS01) by creating, and subsequently removing, TXT records using the Cloudflare API.

CHAPTER 1

Named Arguments

<code>--dns-cloudflare-credentials</code>	Cloudflare <i>credentials</i> INI file. (Required)
<code>--dns-cloudflare-propagation-timeout</code>	The number of seconds to wait for DNS to propagate before asking the ACME server to verify the DNS record. (Default: 10)

Credentials

Use of this plugin requires a configuration file containing Cloudflare API credentials, obtained from your Cloudflare account page.

Listing 1: Example credentials file:

```
# Cloudflare API credentials used by Certbot
dns_cloudflare_email = cloudflare@example.com
dns_cloudflare_api_key = 0123456789abcdef0123456789abcdef01234567
```

The path to this file can be provided interactively or using the `--dns-cloudflare-credentials` command-line argument. Certbot records the path to this file for use during renewal, but does not store the file's contents.

Caution: You should protect these API credentials as you would the password to your Cloudflare account. Users who can read this file can use these credentials to issue arbitrary API calls on your behalf. Users who can cause Certbot to run using these credentials can complete a `dns-01` challenge to acquire new certificates or revoke existing certificates for associated domains, even if those domains aren't being managed by this server.

Certbot will emit a warning if it detects that the credentials file can be accessed by other users on your system. The warning reads "Unsafe permissions on credentials configuration file", followed by the path to the credentials file. This warning will be emitted each time Certbot uses the credentials file, including for renewal, and cannot be silenced except by addressing the issue (e.g., by using a command like `chmod 600` to restrict access to the file).

Examples

Listing 1: To acquire a certificate for `example.com`

```
certbot certonly \  
  --dns-cloudflare \  
  --dns-cloudflare-credentials ~/.secrets/certbot/cloudflare.ini \  
  -d example.com
```

Listing 2: To acquire a single certificate for both `example.com` and `www.example.com`

```
certbot certonly \  
  --dns-cloudflare \  
  --dns-cloudflare-credentials ~/.secrets/certbot/cloudflare.ini \  
  -d example.com \  
  -d www.example.com
```

Listing 3: To acquire a certificate for `example.com`, waiting 60 seconds for DNS propagation

```
certbot certonly \  
  --dns-cloudflare \  
  --dns-cloudflare-credentials ~/.secrets/certbot/cloudflare.ini \  
  --dns-cloudflare-propagation-seconds 60 \  
  -d example.com
```


4.1 certbot_dns_cloudflare.dns_cloudflare

DNS Authenticator for Cloudflare.

class certbot_dns_cloudflare.dns_cloudflare.**Authenticator** (*args, **kwargs)
Bases: certbot.plugins.dns_common.DNSAuthenticator

DNS Authenticator for Cloudflare

This Authenticator uses the Cloudflare API to fulfill a dns-01 challenge.

classmethod **add_parser_arguments** (add)
Add plugin arguments to the CLI argument parser.

NOTE: If some of your flags interact with others, you can use cli.report_config_interaction to register this to ensure values are correctly saved/overridable during renewal.

Parameters **add** (*callable*) – Function that proxies calls to `argparse.ArgumentParser.add_argument` prepending options with unique plugin name prefix.

_setup_credentials ()
Establish credentials, prompting if necessary.

_perform (domain, validation_name, validation)
Performs a dns-01 challenge by creating a DNS TXT record.

Parameters

- **domain** (*str*) – The domain being validated.
- **validation_domain_name** (*str*) – The validation record domain name.
- **validation** (*str*) – The validation record content.

Raises **errors.PluginError** – If the challenge cannot be performed

`_cleanup` (*domain, validation_name, validation*)

Deletes the DNS TXT record which would have been created by `_perform_achall`.

Fails gracefully if no such record exists.

Parameters

- **domain** (*str*) – The domain being validated.
- **validation_domain_name** (*str*) – The validation record domain name.
- **validation** (*str*) – The validation record content.

class `certbot_dns_cloudflare.dns_cloudflare._CloudflareClient` (*email, api_key*)

Bases: `object`

Encapsulates all communication with the Cloudflare API.

add_txt_record (*domain, record_name, record_content, record_ttl*)

Add a TXT record using the supplied information.

Parameters

- **domain** (*str*) – The domain to use to look up the Cloudflare zone.
- **record_name** (*str*) – The record name (typically beginning with ‘_acme-challenge.’).
- **record_content** (*str*) – The record content (typically the challenge validation).
- **record_ttl** (*int*) – The record TTL (number of seconds that the record may be cached).

Raises `certbot.errors.PluginError` – if an error occurs communicating with the Cloudflare API

del_txt_record (*domain, record_name, record_content*)

Delete a TXT record using the supplied information.

Note that both the record’s name and content are used to ensure that similar records created concurrently (e.g., due to concurrent invocations of this plugin) are not deleted.

Failures are logged, but not raised.

Parameters

- **domain** (*str*) – The domain to use to look up the Cloudflare zone.
- **record_name** (*str*) – The record name (typically beginning with ‘_acme-challenge.’).
- **record_content** (*str*) – The record content (typically the challenge validation).

_find_zone_id (*domain*)

Find the `zone_id` for a given domain.

Parameters **domain** (*str*) – The domain for which to find the `zone_id`.

Returns The `zone_id`, if found.

Return type `str`

Raises `certbot.errors.PluginError` – if no `zone_id` is found.

_find_txt_record_id (*zone_id, record_name, record_content*)

Find the `record_id` for a TXT record with the given name and content.

Parameters

- **zone_id** (*str*) – The `zone_id` which contains the record.

- **record_name** (*str*) – The record name (typically beginning with ‘_acme-challenge.’).
- **record_content** (*str*) – The record content (typically the challenge validation).

Returns The record_id, if found.

Return type *str*

CHAPTER 5

Indices and tables

- `genindex`
- `modindex`
- `search`

C

certbot_dns_cloudflare, 1
certbot_dns_cloudflare.dns_cloudflare,
9

Symbols

`_CloudflareClient` (class in `certbot_dns_cloudflare.dns_cloudflare`), 10

`_cleanup()` (`certbot_dns_cloudflare.dns_cloudflare.Authenticator` method), 9

`_find_txt_record_id()` (`certbot_dns_cloudflare.dns_cloudflare._CloudflareClient` method), 10

`_find_zone_id()` (`certbot_dns_cloudflare.dns_cloudflare._CloudflareClient` method), 10

`_perform()` (`certbot_dns_cloudflare.dns_cloudflare.Authenticator` method), 9

`_setup_credentials()` (`certbot_dns_cloudflare.dns_cloudflare.Authenticator` method), 9

A

`add_parser_arguments()` (`certbot_dns_cloudflare.dns_cloudflare.Authenticator` class method), 9

`add_txt_record()` (`certbot_dns_cloudflare.dns_cloudflare._CloudflareClient` method), 10

`Authenticator` (class in `certbot_dns_cloudflare.dns_cloudflare`), 9

C

`certbot_dns_cloudflare` (module), 1

`certbot_dns_cloudflare.dns_cloudflare` (module), 9

D

`del_txt_record()` (`certbot_dns_cloudflare.dns_cloudflare._CloudflareClient` method), 10