
certbot-dns-nsone Documentation

Release 0

Certbot Project

Feb 20, 2019

Contents:

1	Named Arguments	3
2	Credentials	5
3	Examples	7
4	API Documentation	9
4.1	certbot_dns_nsone.dns_nsone	9
5	Indices and tables	11
	Python Module Index	13

The `dns_nsone` plugin automates the process of completing a `dns-01` challenge (DNS01) by creating, and subsequently removing, TXT records using the NS1 API.

CHAPTER 1

Named Arguments

<code>--dns-nsone-credentials</code>	NS1 <i>credentials</i> INI file. (Required)
<code>--dns-nsone-propagation-timeout</code>	The number of seconds to wait for DNS to propagate before asking the ACME server to verify the DNS record. (Default: 30)

Credentials

Use of this plugin requires a configuration file containing NS1 API credentials, obtained from your NS1 [account page](#).

Listing 1: Example credentials file:

```
# NS1 API credentials used by Certbot
dns_nsone_api_key = MDAwMDAwMDAwMDAwMDAw
```

The path to this file can be provided interactively or using the `--dns-nsone-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 NS1 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-nsone \  
  --dns-nsone-credentials ~/.secrets/certbot/nsone.ini \  
  -d example.com
```

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

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


4.1 certbot_dns_nsone.dns_nsone

DNS Authenticator for NS1 DNS.

```
class certbot_dns_nsone.dns_nsone.Authenticator (*args, **kwargs)
    Bases: certbot.plugins.dns_common.DNSAuthenticator
```

DNS Authenticator for NS1

This Authenticator uses the NS1 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_nsone.dns_nsone._NS1LexiconClient` (*api_key, ttl*)

Bases: `certbot.plugins.dns_common_lexicon.LexiconClient`

Encapsulates all communication with the NS1 via Lexicon.

CHAPTER 5

Indices and tables

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

C

`certbot_dns_nsone`, 1

`certbot_dns_nsone.dns_nsone`, 9

Symbols

`_NS1LexiconClient` (class in `certbot_dns_nsone.dns_nsone`), 10

`_cleanup()` (`certbot_dns_nsone.dns_nsone.Authenticator` method), 9

`_perform()` (`certbot_dns_nsone.dns_nsone.Authenticator` method), 9

`_setup_credentials()` (`certbot_dns_nsone.dns_nsone.Authenticator` method), 9

A

`add_parser_arguments()` (`certbot_dns_nsone.dns_nsone.Authenticator` class method), 9

`Authenticator` (class in `certbot_dns_nsone.dns_nsone`), 9

C

`certbot_dns_nsone` (module), 1

`certbot_dns_nsone.dns_nsone` (module), 9