certbot-dns-rfc2136 Documentation

Release 0

Certbot Project

Jun 11, 2026

CONTENTS:

1 Named Arguments **3**

2 Credentials **5**

3 Examples **7**
- 3.1 Sample BIND configuration 7
- 3.2 Special considerations for multiple views in BIND 8

4 API Documentation **11**

5 Indices and tables **13**

Python Module Index **15**

Index **17**

The `dns_rfc2136` plugin automates the process of completing a `dns-01` challenge (`DNS01`) by creating, and subsequently removing, TXT records using RFC 2136 Dynamic Updates.

Note

The plugin is not installed by default. It can be installed by heading to certbot.eff.org, choosing your system and selecting the Wildcard tab.

CHAPTER

ONE

NAMED ARGUMENTS

`--dns-rfc2136-credentia`	RFC 2136 *credentials* INI file. (Required)
`--dns-rfc2136-propagati`	The number of seconds to wait for DNS to propagate before asking the ACME server to verify the DNS record. (Default: 60)

CHAPTER

TWO

CREDENTIALS

Use of this plugin requires a configuration file containing the target DNS server and optional port that supports RFC 2136 Dynamic Updates, the name of the TSIG key, the TSIG key secret itself, the algorithm used if it's different to HMAC-MD5, and optionally whether to sign the initial SOA query.

Listing 1: Example credentials file:

```
# Target DNS server (IPv4 or IPv6 address, not a hostname)
dns_rfc2136_server = 192.0.2.1
# Target DNS port
dns_rfc2136_port = 53
# TSIG key name
dns_rfc2136_name = keyname.
# TSIG key secret
dns_rfc2136_secret = 4q4wM/2I180UXoMyN4INVhJNi8V9BCV+jMw2mXgZw/CSuxUT8C7NKKFs␣
↪AmKd7ak51vWKgSl12ib86oQRPkpDjg==
# TSIG key algorithm
dns_rfc2136_algorithm = HMAC-SHA512
# TSIG sign SOA query (optional, default: false)
dns_rfc2136_sign_query = false
```

The path to this file can be provided interactively or using the `--dns-rfc2136-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 this TSIG key material as it can be used to potentially add, update, or delete any record in the target DNS server. 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).

CHAPTER

THREE

EXAMPLES

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

```
certbot certonly \
  --dns-rfc2136 \
  --dns-rfc2136-credentials ~/.secrets/certbot/rfc2136.ini \
  -d example.com
```

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

```
certbot certonly \
  --dns-rfc2136 \
  --dns-rfc2136-credentials ~/.secrets/certbot/rfc2136.ini \
  -d example.com \
  -d www.example.com
```

Listing 3: To acquire a certificate for `example.com`, waiting 30 seconds for DNS propagation

```
certbot certonly \
  --dns-rfc2136 \
  --dns-rfc2136-credentials ~/.secrets/certbot/rfc2136.ini \
  --dns-rfc2136-propagation-seconds 30 \
  -d example.com
```

3.1 Sample BIND configuration

Here's a sample BIND configuration for Certbot to use. You will need to generate a new TSIG key, include it in the BIND configuration and grant it permission to issue updates on the target DNS zone.

Listing 4: Generate a new SHA512 TSIG key

```
tsig-keygen -a HMAC-SHA512 keyname.
```

Note

Prior to BIND version 9.10.0, you will need to use `dnssec-keygen` to generate TSIG keys. Try and use the most secure algorithm supported by your DNS server.

Listing 5: Sample BIND configuration

```
key "keyname." {
  algorithm hmac-sha512;
  secret "4q4wM/2I180UXoMyN4INVhJNi8V9BCV+jMw2mXgZw/CSuxUT8C7NKKFs␣
↪AmKd7ak51vWKgSl12ib86oQRPkpDjg==";
};

zone "example.com." IN {
  type master;
  file "named.example.com";
  update-policy {
    grant keyname. name _acme-challenge.example.com. txt;
  };
};
```

Note

This configuration limits the scope of the TSIG key to just be able to add and remove TXT records for one specific host for the purpose of completing the `dns-01` challenge. If your version of BIND doesn't support the `update-policy` directive, then you can use the less-secure `allow-update` directive instead. See the BIND documentation for details.

3.2 Special considerations for multiple views in BIND

If your BIND configuration leverages multiple views, Certbot may fail with an `Unable to determine base domain for _acme-challenge.example.com` error. This error occurs when Certbot isn't able to communicate with an authoritative nameserver for the zone, one that answers with the AA (Authoritative Answer) flag set in the response.

A common multiple view configuration with two views, external and internal, can cause this error. If the zone is only present in the external view, and the *credentials* `dns_rfc2136_server` setting is set (e.g. 127.0.0.1) so the DNS server's `match-clients` view option causes the DNS server to route Certbot's query to the internal view; the internal view doesn't contain the zone, so the response won't have the AA flag set.

One solution is to logically place the zone into the view Certbot is sending queries to, with an in-view zone option. The zone will be then visible in both zones with exactly the same content.

Note

Order matters in BIND views: the `in-view` zone option must refer to a view defined preceding it. It cannot refer to a view defined later in the configuration file.

Listing 6: Split-view BIND configuration

```
key "keyname." {
  algorithm hmac-sha512;
  secret "4q4wM/2I180UXoMyN4INVhJNi8V9BCV+jMw2mXgZw/CSuxUT8C7NKKFs␣
↪AmKd7ak51vWKgSl12ib86oQRPkpDjg==";
};
```

(continues on next page)

(continued from previous page)

```
// adjust internal-addresses to suit your needs
acl internal-address { 127.0.0.0/8; 10.0.0.0/8; 192.168.0.0/16; 172.16.0.0/12; };

view "external" {
  match-clients { !internal-addresses; any; };

  zone "example.com." IN {
    type master;
    file "named.example.com";
    update-policy {
      grant keyname. name _acme-challenge.example.com. txt;
    };
  };
};

view "internal" {
  zone "example.com." IN {
    in-view external;
  };
};
```

Another solution is to add `dns_rfc2136_sign_query = true` to the configuration file and then add the key to the `match-clients` list within the external zone view. All queries signed with this key should then be directed to this view, regardless of source IP.

Listing 7: Split-view BIND configuration with key-based ACLs

```
key "keyname." {
  algorithm hmac-sha512;
  secret "4q4wM/2I180UXoMyN4INVhJNi8V9BCV+jMw2mXgZw/CSuxUT8C7NKKFs␣
↪AmKd7ak51vWKgSl12ib86oQRPkpDjg==";
};

// adjust internal-addresses to suit your needs
acl internal-address { 127.0.0.0/8; 10.0.0.0/8; 192.168.0.0/16; 172.16.0.0/12; };

acl certbot-keys { key keyname.; }

view "external" {
  match-clients { acl certbot-keys; !internal-addresses; any; };

  zone "example.com." IN {
    type master;
    file "named.example.com";
    update-policy {
      grant keyname. name _acme-challenge.example.com. txt;
    };
  };
};
```

CHAPTER

FOUR

API DOCUMENTATION

Certbot plugins implement the Certbot plugins API, and do not otherwise have an external API.

CHAPTER

FIVE

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

C

certbot_dns_rfc2136, 1

INDEX

C

certbot_dns_rfc2136
    module, 1

M

module
    certbot_dns_rfc2136, 1