

---

# **certbot-dns-route53 Documentation**

*Release 0*

**Certbot Project**

**Oct 14, 2021**



---

## Contents:

---

<b>1</b>	<b>Named Arguments</b>	<b>3</b>
<b>2</b>	<b>Credentials</b>	<b>5</b>
<b>3</b>	<b>Examples</b>	<b>7</b>
<b>4</b>	<b>API Documentation</b>	<b>9</b>
<b>5</b>	<b>Indices and tables</b>	<b>11</b>
	<b>Python Module Index</b>	<b>13</b>
	<b>Index</b>	<b>15</b>



The `dns_route53` plugin automates the process of completing a `dns-01` challenge (DNS01) by creating, and subsequently removing, TXT records using the Amazon Web Services Route 53 API.

---

**Note:** The plugin is not installed by default. It can be installed by heading to [certbot.eff.org](https://certbot.eff.org), choosing your system and selecting the Wildcard tab.

---



# CHAPTER 1

---

## Named Arguments

---

<code>--dns-route53-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)
--	--





Use of this plugin requires a configuration file containing Amazon Web Services API credentials for an account with the following permissions:

- `route53:ListHostedZones`
- `route53:GetChange`
- `route53:ChangeResourceRecordSets`

These permissions can be captured in an AWS policy like the one below. Amazon provides [information about managing access](#) and [information about the required permissions](#)

Listing 1: Example AWS policy file:

```
{
  "Version": "2012-10-17",
  "Id": "certbot-dns-route53 sample policy",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "route53:ListHostedZones",
        "route53:GetChange"
      ],
      "Resource": [
        "*"
      ]
    },
    {
      "Effect" : "Allow",
      "Action" : [
        "route53:ChangeResourceRecordSets"
      ],
      "Resource" : [
        "arn:aws:route53:::hostedzone/YOURHOSTEDZONEID"
      ]
    }
  ]
}
```

(continues on next page)

(continued from previous page)

```
}  
  ]  
}
```

The [access keys](#) for an account with these permissions must be supplied in one of the following ways, which are discussed in more detail in the Boto3 library's documentation about [configuring credentials](#).

- Using the `AWS_ACCESS_KEY_ID` and `AWS_SECRET_ACCESS_KEY` environment variables.
- Using a credentials configuration file at the default location, `~/.aws/config`.
- Using a credentials configuration file at a path supplied using the `AWS_CONFIG_FILE` environment variable.

Listing 2: Example credentials config file:

```
[default]  
aws_access_key_id=AKIAIOSFODNN7EXAMPLE  
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
```

**Caution:** You should protect these API credentials as you would a password. Users who can read this file can use these credentials to issue some types of API calls on your behalf, limited by the permissions assigned to the account. 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 domains these credentials are authorized to manage.

---

## Examples

---

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

```
certbot certonly \  
  --dns-route53 \  
  -d example.com
```

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

```
certbot certonly \  
  --dns-route53 \  
  -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-route53 \  
  --dns-route53-propagation-seconds 30 \  
  -d example.com
```



## CHAPTER 4

---

### API Documentation

---

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



## CHAPTER 5

---

### Indices and tables

---

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





**C**

`certbot_dns_route53, 1`



## C

certbot\_dns\_route53 (*module*), 1