ICANN Centralized Zone Data Service REST API (CZDS REST API)

Version 1.0.6
2019-01-30

Document Revision History

Version	Publishing date	Description of the change	Projected date to start with the version of the specification in production
1.0	2018/10/19	First version.	
1.0.1	2018/11/26	Fix the authenticate curl example on page 5 and getting zone file `gzip` parameter on page 8.	
1.0.2	2018/11/28	Update the description on zone file compression in Section 5.2	
1.0.3	2018/11/29	Update Session 5.2 to remove "`Accept-Encoding: gzip`". The zone file from the server is already compressed. It is not necessary to specify `gzip` accept-encoding.	
1.0.4	2018/12/04	Add a new section - 5.3 Checking zone file size. Update `access_token` lifetime to 24 hours.	
1.0.5	2019/01/22	Add production URLs in Section 2. Add User-Agent Header Section 3.1, 5.1 and 5.2 Add filename in Section 5.3.	
1.0.6	2019/01/30	Add `rate limit` on `/api/authenticate` endpoint in Section 3.1	

1. Introduction

This document describes the REST API provided by ICANN to the registered Centralized Zone Date Service (CZDS) users to download their authorized zone files.

1.1. Credentials

The CZDS Authentication API uses the same credentials as the CZDS portal site to authenticate the users. However, there is no REST API for managing the user's credentials. Any credential management operation, such as, resetting password, must go through the CZDS portal site.

1.2. Internet Protocol

The CZDS REST API supports IPv4 transport only.

1.3. Secure Communication

In order to provide confidentiality, server authentication, and integrity in the communication channel, only HTTPS is available for CZDS REST API.

2. Common elements used in this specification

In the following section, common elements used in this specification are described:

- **<auth_url>**: the URL where a registered user can obtain a token that is requires by CZDS REST API.

  Note: the URL for the Testbed: https://account-api-test.icann.org/api/authenticate
  Note: the URL for Production: https://account-api.icann.org/api/authenticate

- **<username>:** the `username` registered on CZDS.

- **<password>:** the `password` credential for the user.

- **<base_url>:** the base URL of the CZDS REST API.

  Note: the URL for the Testbed: https://czds-api-test.icann.org
  Note: the URL for the Production: https://czds-api.icann.org

- **<accessToken>:** A JSON Web token (JWT) as described in RFC 7519. A JWT is represented as a sequence of URL-safe parts separated by period ('.') characters. Each part contains a base64url-encoded value.

The `<accessToken>` can be obtained by invoking the authentication API with valid CZDS user credential. It is required to perform operations on CZDS REST API to list download links and download zone files.

- **<tld>:** Name of the Top-Level Domain or zone.
- **<exp_claim_value>**: The value of the `exp` claim of `<accessToken>` that indicates the lifetime of the token.

3. Authentication

The CZDS REST API is based on Oauth 2.0 Authorization framework as described in RFC 6749. This section describes how to obtain an Access Token with the user's credential. The credential is only needed when requesting a token. The rest of the API calls require the access token obtained here.

3.1. Obtaining a token

`POST <auth_url>`

Obtains an access token.

Parameters:

The following parameters must be provided in the request body:

- "username": "<value>"

Username registered in the CZDS

- "password": "<value>"

Password for username registered in the CZDS

The following headers must be provided in POST Method:

- "Content-Type: application/json"
- "Accept: application/json"
- "User-Agent: `<product> / <product-version> <comment>`"
  You must provide a valid User-Agent string. Otherwise, you might be redirected to an ICANN maintenance page. Please check with the HTTP client or library you're using to see if it provides a default User-Agent header. If not, please make sure you provide one.

Possible results:

- HTTP 200, when a valid request is received (`username` and `password` from the CZDS are correct), the API method `<auth_url>` sets HTTP header Content-type to

"application/json", returns HTTP 200 code and provides a JSON object in the body with following keys:

"accessToken": "<value>" where value is a JWT token (A sequence of URL-safe parts separated by period ('.') characters

- HTTP 400, the <auth_url> API method provides an HTTP 400 (Bad Request) status code when the request is malformed.

- HTTP 401, the <auth_url> API method provides an HTTP 401 status code when invalid credentials are provided. There is no text response in the body.

- HTTP 415, when unsupported content type header is sent, the API method <auth_url> sets HTTP header Content-type to "application/json", returns HTTP 415 code and provides a JSON object with following keys:

    - "timestamp": text value with timestamp of transaction
    - "status": 415
    - "error": Unsupported media type
    - "message" text value with error message
    - "path" text value with path from URL

- HTTP 429, too many authentication attempts from the same IP address. The rate limit is IP address-based. It allows 8 authentication attempts in 5 minutes. If the rate limit is reached, any subsequent authentication requests from the same IP address will be denied for the remainder of the 5-minute cycle.

- HTTP 500, the <auth_url> API method provides an HTTP 500 status code when an Internal Server Error is found.

Example using curl (https://curl.haxx.se/) for a authentication request in TestBed:

Request Example:

```
curl -i -X POST \
-H "Accept: application/json" \
-H "Content-Type: application/json" \
-d '{
     "username": "cindy.riley41@example.com",
     "password": "tlpWtpkEj8NT2#m"
}' https://account-api-test.icann.org/api/authenticate
```

Response Example:

```
{
    "accessToken":
"eyJraWQiOiI5UUR4ci16dHlLbjlHVEE0LS1HR3FqQ3dUVk1Fc0MxRXJNb3VtVlNEM1NBIiwiYWxnIjoiUlMyN
TYifQ.eyJ2ZXIiOjEsImp0aSI6IkFULjYwdGd0MXpJQWllRkdoUFRsSTRTUlprRHpQVkNVT0MyalFTSi1mM1p0
VVUiLCJpc3MiOiJodHRwczovL2Rldi05ODcyNDkub2t0YXByZXZpZXcuY29tL29hdXRoMi9hdXNnaW42aDZ2Zz
JxdkliUTBoNyIsImF1ZCI6Imh0dHBzOi8vYXBpdGVzdGJlZC5pY2Fubi5vcmciLCJpYXQiOjE1NDIzMDc2MjQs
ImV4cCI6MTU0MjMxMTIyNCwiY2lkIjoiMG9hZ25lcTc0M2Q0T1BVVVgwaDciLCJ1aWQiOiIwMHVnaWpxeDhrcF
JEb1RVRTBoNyIsInNjcCI6WyJpY2Fubi1jdXN0b20iLCJvcGVuaWQiXSwic3ViIjoiZG9uZ21laS5jYW9AaWNh
bm4ub3JnIiwiZ2l2ZW5fbmFtZSI6IkRvbmdtZWkiLCJmYW1pbHlfbmFtZSI6IkNhbyIsImVtYWlsIjoiZG9uZ2
1laS5jYW9AaWNhbm4ub3JnIn0.NJtVnGTtpzJOcLzwnw-eRjL15W5ytzgL7-ZQ5tYypMuFxYp_wwgMq2tKGTi8
```

```
OQdCaw9JkVzGijVRvSUguM_aIplON5uQZIBCR59wRqPoK6H0S1Ds3IKLQSsYxRbZXwW3MXmuw1lqocBfNFwLdm
UNmmuJflRsLK4WO-OWmuOtRJPReJ3jbLv67XPNKauiQaHSvLa-2rQrpEddNxDnJuaUoZ10m3bllmpUqlQR2JRB
Sr5kddsJ1uIWLrJOin-IRQEx0EyS0zOmCUeLY8N46VRmzr5WMpm0dLgrvOZRZTNK9NusP6aG6vjdTGLcIz6GHn
4gdPv19s_H90ROd2cWehVoVQ"
}
```


Note: Every successful authentication API call will generate a brand new token.

3.2. Token Expiration

The `<accessToken>` obtained by the authentication API call is invalid after its expiration time, which is 24 hours after the token is created.

The expiration time of the `<accessToken>` is specified by the `exp` (Expiration time) Claim. JWT claims are described in section 4 of RFC 7519.

The `exp` Claim from `<accessToken>` can be obtained by following next steps:

1. Get a new `<accessToken>` using the `<auth_url>` of the CZDS RST API.

   Example using curl (https://curl.haxx.se/) for a login request in TestBed, and using sed and awk commands for output formatting:

   ```
   access_token=`curl -d '{"username":"<username>","password":"<password>"}' -s
   -H "Content-Type: application/json" -X POST
   https://accounts-api-test.icann.org/api/authenticate | sed "s/\"//g" |sed
   "s/:/ /g" | awk '{print $2}' | sed 's/}//g'`
   ```

   Note. The value of the `<accessToken>` is stored on access_token variable.

2. Obtain JWT payload from the `<accessToken>`

   Example using sed and awk commands for output formatting:

   ```
   jwt_payload=`echo ${access_token} | sed "s/\./ /g" | awk '{print $2}'`
   ```

   Note. The second URL-safe part of the `<accessToken>` is obtained and stored on jwt_payload variable

3. Decode base64url value obtained from JWT payload.

   Example using node.js base64-url-cli (https://www.npmjs.com/package/base64-url-cli) for base64-url decoding.

   ```
   base64url decode ${jwt_payload}
   ```

   Note. The command will provide a JSON object where "`exp`" Claim can be parsed.

4. Convert datetime value from `exp` key from JSON object.

   Example using GNU date:

```
date -d @<exp_claim_value>
```

4. Authorization

When sending a request to the CZDS REST API, the client must set HTTP header "Authorization: Bearer `<accessToken>`", where `<accessToken>` is the token obtained in the authentication API call described in the previous section.

The following response may be returned by the API call:

- HTTP/401, when no valid bearer token `<accessToken>`) is provided, the API method provides an HTTP/401 status code, sets the HTTP header Content-type to "`text/dns`". No text response is provided in the "HTTP body:".

5. Getting zone(s) information

This section describes the REST API to access the zone information in CZDS.

5.1. Listing zone download links

```
GET <base_url>/czds/downloads/links
```

Lists all the zone file download links that are authorized for the user.

Parameters:

The following headers must be provided in GET Method:

- "Authorization: Bearer `<accessToken>`"
- "User-Agent: `<product> / <product-version> <comment>`"
  You must provide a valid User-Agent string. Otherwise, you might be redirected to an ICANN maintenance page. Please check with the HTTP client or library you're using to see if it provides a default User-Agent header. If not, please make sure you provide one.

Possible results:

HTTP/200, when a valid request is received, the `<base_url>/czds/downloads/links` API method provides an HTTP/200 status code and sets the HTTP header Content-type to "`application/json;charset=UTF-8`".

If a valid request is received, a JSON array with a list of authorized download zone URLs is returned in the HTTP body. For example:

```
[
   "<base_url>/czds/downloads/<tld>.zone",
   "<base_url>/czds/downloads/<tld>.zone"
```

```
]
```

Each URL in this list can be used to retrieve the respective zone file.

- HTTP 400, the `<base_url>` API method provides an HTTP 400 (Bad Request) status code when the request is malformed.

- HTTP 500, the `<base_url>/czds/downloads/links` API method provides an HTTP 500 status code when an Internal Server Error is found.

Example using curl to request list of links for authorized zones in TestBed:

Request Example:

```
curl -i -X GET \
-H "Accept: application/json" \
-H "Content-Type: application/json" \
-H "Authorization: Bearer
eyJraWQiOiI5UUR4ci16dHlLbjlHVEE0LS1HR3FqQ3dUVk1Fc0MxRXJNb3VtVlNEM1NBIiwiYWxnIjoiUlMyNT
YifQ.eyJ2ZXIiOjEsImp0aSI6IkFULjYwdGd0MXpJQWllRkdoUFRsSTRTUlprRHpQVkNVT0MyalFTSi1mM1p0V
VUiLCJpc3MiOiJodHRwczovL2Rldi05ODcyNDkub2t0YXByZXZpZXcuY29tL29hdXRoMi9hdXNnaW42aDZ2ZzJ
xdkliUTBoNyIsImF1ZCI6Imh0dHBzOi8vYXBpdGVzdGJlZC5pY2Fubi5vcmciLCJpYXQiOjE1NDIzMDc2MjQsI
mV4cCI6MTU0MjMxMTIyNCwiY2lkIjoiMG9hZ25lcTc0M2Q0T1BVVVgwaDciLCJ1aWQiOiIwMHVnaWpxeDhrcFJ
Eb1RVRTBoNyIsInNjcCI6WyJpY2Fubi1jdXN0b20iLCJvcGVuaWQiXSwic3ViIjoiZG9uZ21laS5jYW9AaWNhb
m4ub3JnIiwiZ2l2ZW5fbmFtZSI6IkRvbmdtZWkiLCJmYW1pbHlfbmFtZSI6IkNhbyIsImVtYWlsIjoiZG9uZ21
laS5jYW9AaWNhbm4ub3JnIn0.NJtVnGTtpzJOcLzwnw-eRjL15W5ytzgL7-ZQ5tYypMuFxYp_wwgMq2tKGTi8O
QdCaw9JkVzGijVRvSUguM_aIplON5uQZIBCR59wRqPoK6H0S1Ds3IKLQSsYxRbZXwW3MXmuw1lqocBfNFwLdmU
NmmuJflRsLK4WO-OWmuOtRJPReJ3jbLv67XPNKauiQaHSvLa-2rQrpEddNxDnJuaUoZ10m3bllmpUqlQR2JRBS
r5kddsJ1uIWLrJOin-IRQEx0EyS0zOmCUeLY8N46VRmzr5WMpm0dLgrvOZRZTNK9NusP6aG6vjdTGLcIz6GHn4
gdPv19s_H90ROd2cWehVoVQ" \
https://czds-api-test.icann.org/czds/downloads/links
```

Response Example:

```
[
    "https://czds-api-test.icann.org/czds/downloads/example1.zone",
    "https://czds-api-test.icann.org/czds/downloads/example2.zone"
]
```

5.2. Getting a zone file

`GET <base_url>/czds/downloads/<tld>.zone`

Download the zone file for the given TLD. The downloaded file is in `gzip` format.

Parameters:

The following header must be provided in the GET Method:

- "Authorization: Bearer `<accessToken>`"

- "User-Agent: `<product> / <product-version> <comment>`"
  You must provide a valid User-Agent string. Otherwise, you might be redirected to an ICANN maintenance page. Please check with the HTTP client or library you're using to see if it provides a default User-Agent header. If not, please make sure you provide one.

Possible results:

- HTTP/200, when a valid request is received, the `<base_url>/czds/downloads/<tld>.zone` API method provides an HTTP/200 status code, sets the HTTP header Content-type to "`text/dns`" and provides in the HTTP body the contents of the zone file in `gzip` format.

- HTTP 400, the `<auth_url>`API method provides an HTTP 400 (Bad Request) status code when the request is malformed.

- HTTP/403, when the requested zone does not exist or the username that generated the `<accessToken>` has no permissions to download that zone in the CZDS system, the API method `<base_url>/czds/downloads/<tld>.zone` provides an HTTP/403 status code and sets the HTTP header Content-type to "`text/dns`". No text response is provided in the "HTTP body:".

- HTTP/409, the `<base_url>/czds/downloads/<tld>.zone` API method provides an HTTP/409 status code when the username that generated `<accessToken>` has not accepted the new Terms & Condition on the CZDS system. The user will need to log in the CZDS portal to accept Terms & Conditions before use this API method.

Examples using CURL to request a zone file in TestBed:

Request Example:

```
curl -i -X GET \
-H "Authorization: Bearer
eyJraWQiOiI5UUR4ci16dHlLbjlHVEE0LS1HR3FqQ3dUVk1Fc0MxRXJNb3VtVlNEM1NBIiwiYWxnIjoiUlMyNT
YifQ.eyJ2ZXIiOjEsImp0aSI6IkFULjYwdGd0MXpJQWllRkdoUFRsSTRTUlprRHpQVkNVT0MyalFTSi1mM1p0V
VUiLCJpc3MiOiJodHRwczovL2Rldi05ODcyNDkub2t0YXByZXZpZXcuY29tL29hdXRoMi9hdXNnaW42aDZ2ZzJ
xdkliUTBoNyIsImF1ZCI6Imh0dHBzOi8vYXBpdGVzdGJlZC5pY2Fubi5vcmciLCJpYXQiOjE1NDIzMDc2MjQsI
mV4cCI6MTU0MjMxMTIyNCwiY2lkIjoiMG9hZ25lcTc0M2Q0T1BVVVgwaDciLCJ1aWQiOiIwMHVnaWpxeDhrcFJ
Eb1RVRTBoNyIsInNjcCI6WyJpY2Fubi1jdXN0b20iLCJvcGVuaWQiXSwic3ViIjoiZG9uZ21laS5jYW9AaWNhb
m4ub3JnIiwiZ2l2ZW5fbmFtZSI6IkRvbmdtZWkiLCJmYW1pbHlfbmFtZSI6IkNhbyIsImVtYWlsIjoiZG9uZ21
laS5jYW9AaWNhbm4ub3JnIn0.NJtVnGTtpzJOcLzwnw-eRjL15W5ytzgL7-ZQ5tYypMuFxYp_wwgMq2tKGTi8O
QdCaw9JkVzGijVRvSUguM_aIplON5uQZIBCR59wRqPoK6H0S1Ds3IKLQSsYxRbZXwW3MXmuw1lqocBfNFwLdmU
NmmuJflRsLK4WO-OWmuOtRJPReJ3jbLv67XPNKauiQaHSvLa-2rQrpEddNxDnJuaUoZ10m3bllmpUqlQR2JRBS
r5kddsJ1uIWLrJOin-IRQEx0EyS0zOmCUeLY8N46VRmzr5WMpm0dLgrvOZRZTNK9NusP6aG6vjdTGLcIz6GHn4
gdPv19s_H90ROd2cWehVoVQ" \
https://czds-api-test.icann.org/czds/downloads/example1.zone >
/tmp/example1.zone.txt.gz
```

Note: the download file is in `gzip` format.

5.3. Checking zone file size and filename

```
HEAD <base_url>/czds/downloads/<tld>.zone
```

Checking a zone file size and filename without downloading the zone file. The `Content-Length` header indicates the size of the zone file in bytes. The default filename for the zone file can be found in the `Content-Disposition` header.

Parameters:

Same parameters are required as in 5.2 Getting a zone file.

Possible results:

Same possible results should be expected as in 5.2 Getting a zone file.

Examples using CURL to get the Content-Length header in TestBed:

Request Example:

```
curl -I \
-H "Authorization: Bearer
eyJraWQiOiI5UUR4ci16dHlLbjlHVEE0LS1HR3FqQ3dUVk1Fc0MxRXJNb3VtVlNEM1NBIiwiYWxnIjoiUlMyNT
YifQ.eyJ2ZXIiOjEsImp0aSI6IkFULjYwdGd0MXpJQWllRkdoUFRsSTRTUlprRHpQVkNVT0MyalFTSi1mM1p0V
VUiLCJpc3MiOiJodHRwczovL2Rldi05ODcyNDkub2t0YXByZXZpZXcuY29tL29hdXRoMi9hdXNnaW42aDZ2ZzJ
xdkliUTBoNyIsImF1ZCI6Imh0dHBzOi8vYXBpdGVzdGJlZC5pY2Fubi5vcmciLCJpYXQiOjE1NDIzMDc2MjQsI
mV4cCI6MTU0MjMxMTIyNCwiY2lkIjoiMG9hZ25lcTc0M2Q0T1BVVVgwaDciLCJ1aWQiOiIwMHVnaWpxeDhrcFJ
Eb1RVRTBoNyIsInNjcCI6WyJpY2Fubi1jdXN0b20iLCJvcGVuaWQiXSwic3ViIjoiZG9uZ21laS5jYW9AaWNhb
m4ub3JnIiwiZ2l2ZW5fbmFtZSI6IkRvbmdtZWkiLCJmYW1pbHlfbmFtZSI6IkNhbyIsImVtYWlsIjoiZG9uZ21
laS5jYW9AaWNhbm4ub3JnIn0.NJtVnGTtpzJOcLzwnw-eRjL15W5ytzgL7-ZQ5tYypMuFxYp_wwgMq2tKGTi8O
QdCaw9JkVzGijVRvSUguM_aIplON5uQZIBCR59wRqPoK6H0S1Ds3IKLQSsYxRbZXwW3MXmuw1lqocBfNFwLdmU
NmmuJflRsLK4WO-OWmuOtRJPReJ3jbLv67XPNKauiQaHSvLa-2rQrpEddNxDnJuaUoZ10m3bllmpUqlQR2JRBS
r5kddsJ1uIWLrJOin-IRQEx0EyS0zOmCUeLY8N46VRmzr5WMpm0dLgrvOZRZTNK9NusP6aG6vjdTGLcIz6GHn4
gdPv19s_H90ROd2cWehVoVQ" \
https://czds-api-test.icann.org/czds/downloads/example1.zone
```

Response Header Example (partial):

```
…
Content-Type: application/x-gzip
Content-Disposition: attachment;filename=example1.txt.gz
Content-Length: 3716
…
```

Note: the Content-Length is in decimal number of bytes.