Black Box DCX3000 / DCX1000
Using the API

updated 2/22/2017

This document will give you a brief overview of how to access the DCX3000 / DCX1000 API and how you can interact with it using an online tool. It will not however go into the inner workings.

The API on the DCX technology adheres to RESTful and CORS standards, and can be downloaded from the DCX itself by following the URL:

http(s)://<IP ADDRESS>/rest-api

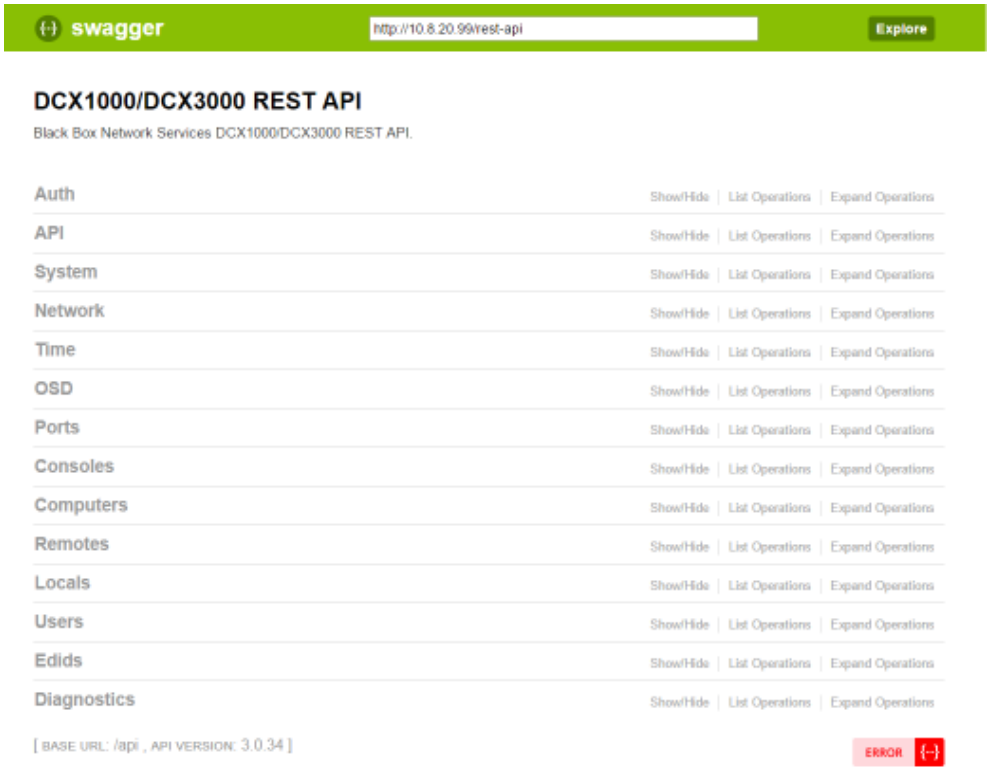
It is presented in YAML format, which is a human-readable data serialization language.

YAML Tool

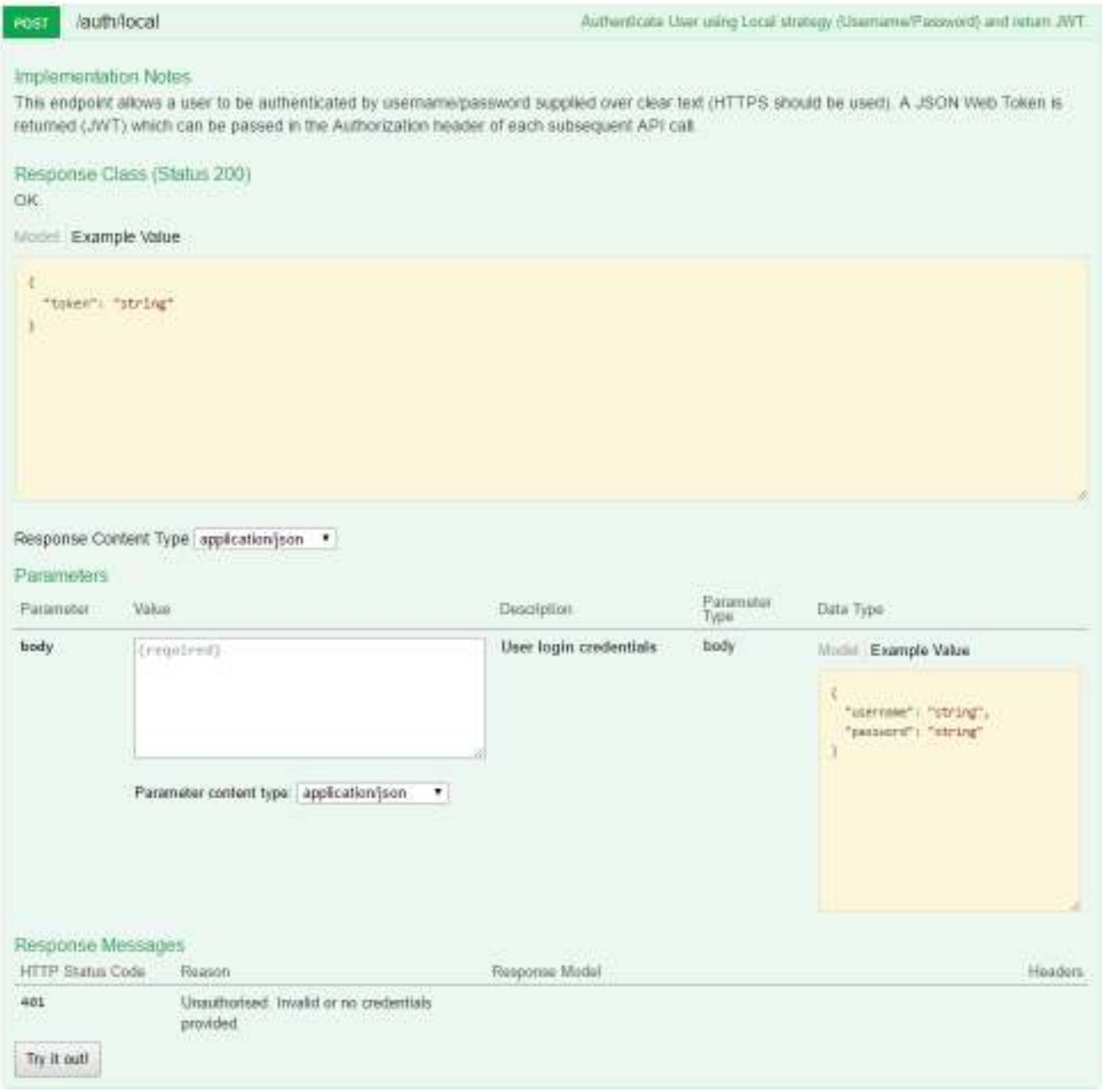
There is an online tool called Swagger that allows you to view the API in an easy format and interact with the API. The DDX requires internet access in order to achieve this.

Open a web browser and go to the Swagger online demo: http://petstore.swagger.io

At the top of the page, enter the URL of the DCX switch, including the path to the reset-api. Press Explore (i.e. http://10.0.10.14/rest-api)



You will be presented with a list of API functions on the DCX. In order to perform any function, you need a token which is provided when you authenticate with the DCX. To acquire the token, click on Expand operations for the Auth function.



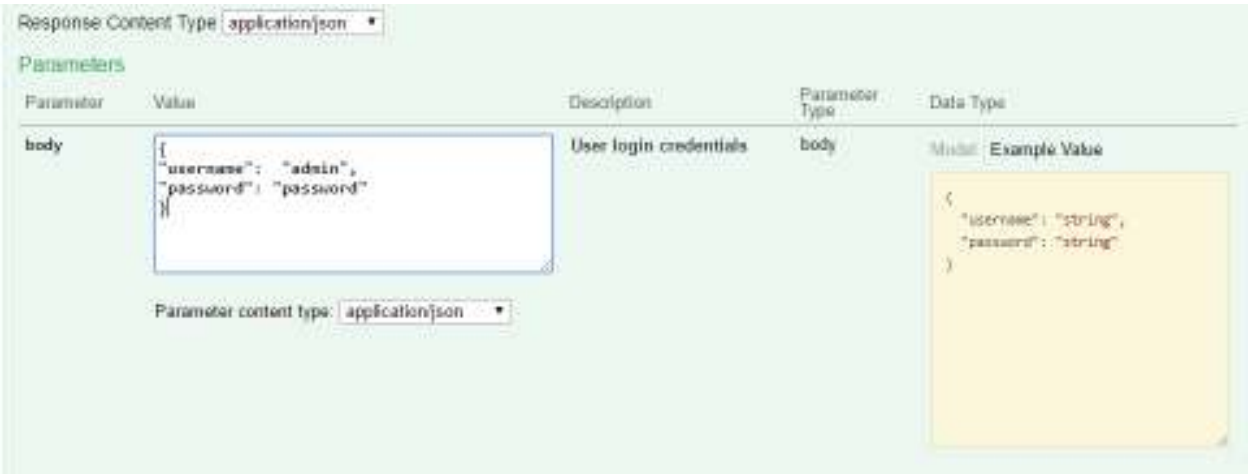
In the Body section under parameters, enter the username and password for the DCX in the following format.

{

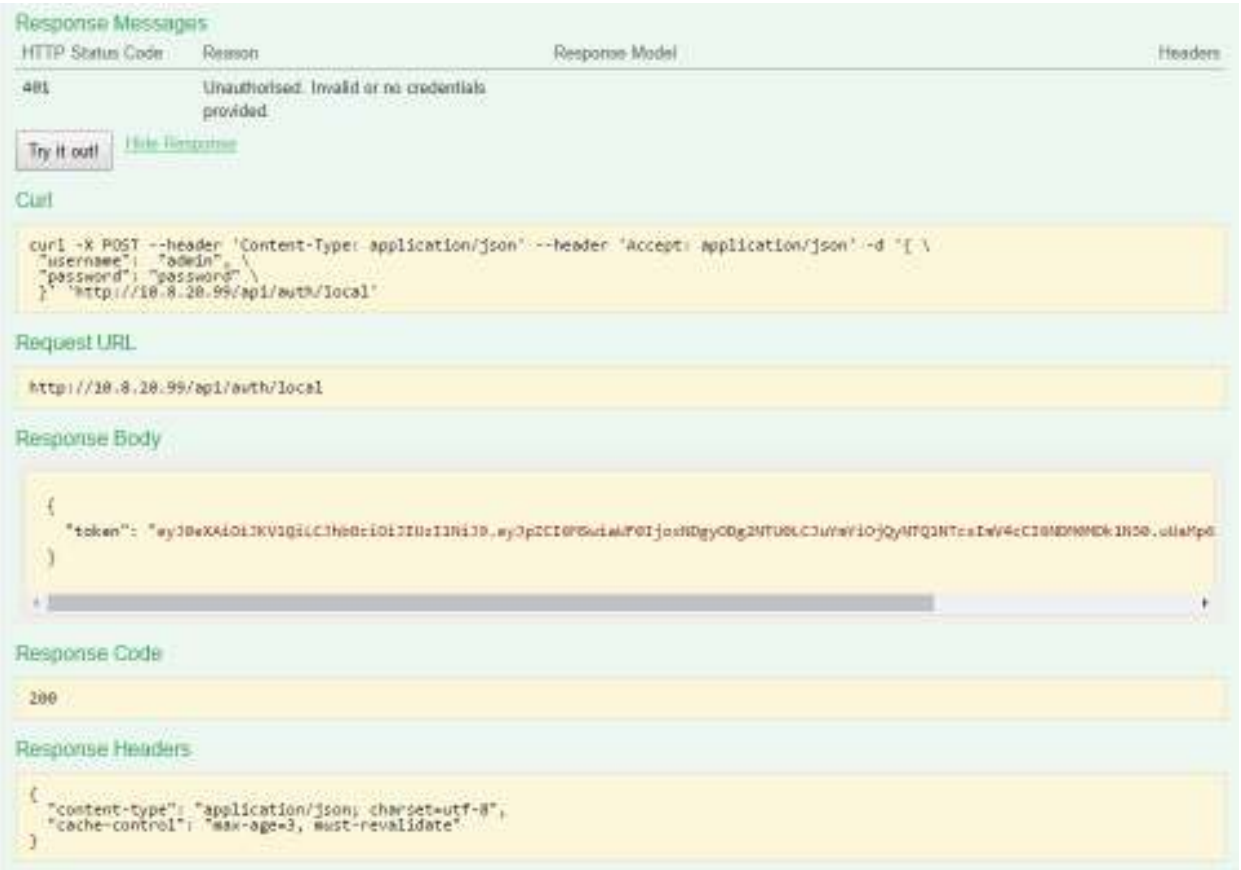
“username”: “username”

“password”: “password”

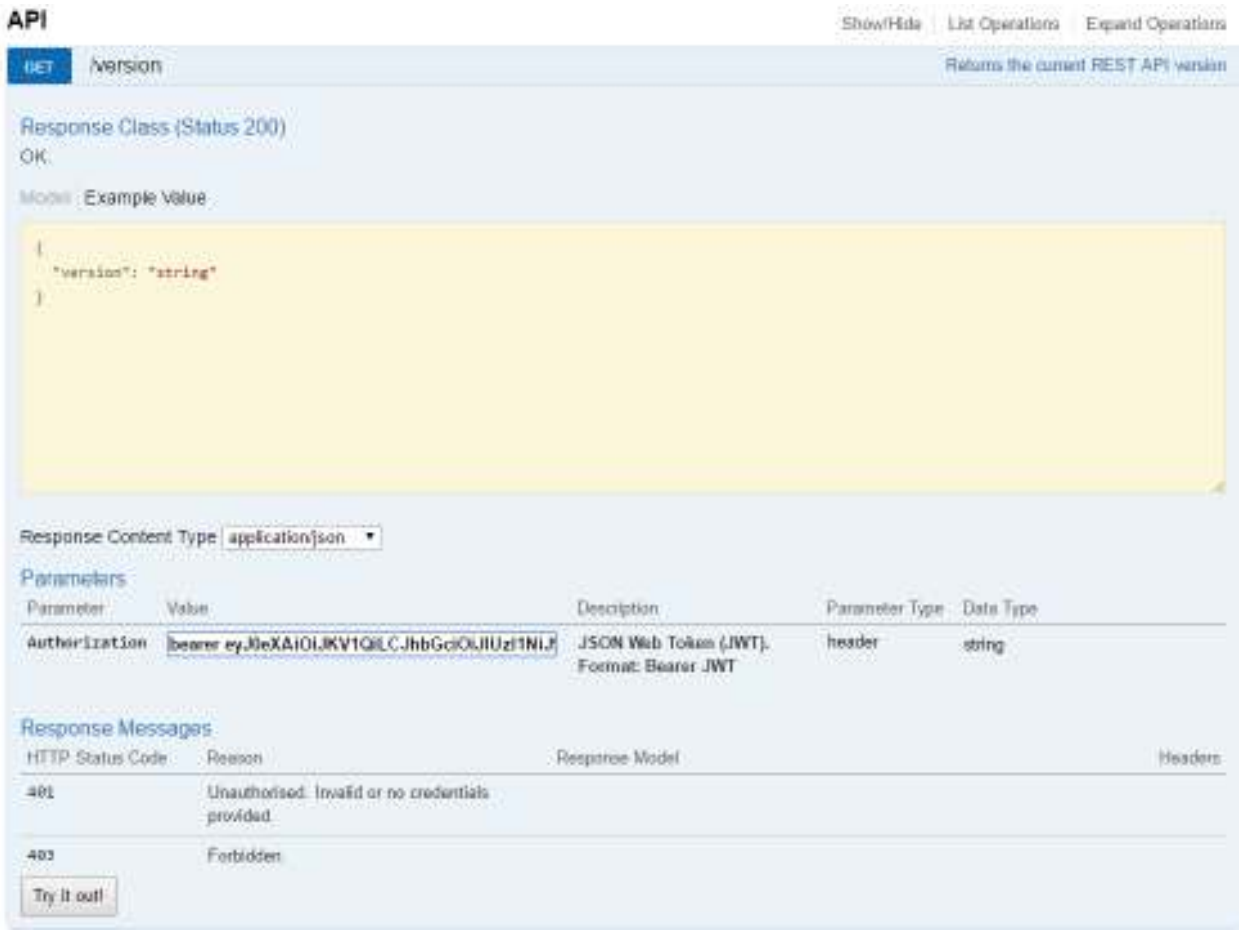
}



Press the “Try it Out!” button to execute the function. In the response body, you should receive a response showing your unique session token.



Now that you have a token, you can use any of the available functions. For example, to see the API version, expand the API operations and in the Authorization field under Parameters, enter the following format: bearer <token>



Press the “Try it Out!” button to execute the function. The response should look something like this…

Raw Connection Example

Using Telnet and port 80, you can issue the HTML/JSON commands in a RAW form to the DCX. If you are writing a controller application, you will need to ensure it can handle HTML requests and parse the JSON responses appropriately.

A POST or PUT request must include the content length in the header as you would for any web service. The content length is the number of bytes (characters) in the body/payload of the request. If this number is not correct, you will either be disconnected form the DCX or receive an error.

The first function that you must perform is authentication with DCX to generate a token by sending your Admin login credentials that you use to log into the web interface. The token does not include the double quotes (*).

Request:

```
POST /api/auth/local HTTP/1.1
Host: 10.0.10.14
Content-Type: application/json;
Content-Length: 43
{"username":"admin","password":"password"}
```

Response:

```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Cache-Control: max-age=3, must-revalidate
App-Version: 0.3.94
Content-Type: application/json; charset=utf-8
Content-Length: 164
Vary: Accept-Encoding
Date: Wed, 09 Nov 2016 12:34:20 GMT
Connection: keep-alive
{"token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpZCI6MSwiaWF0IjoxNDc4Njk0ODYwLCJuYmYiOjI4NDg5ODQsImV4cCI6MjkzNTM4NH0.qi5JpApGP8Tcttvw4IW6FK9ldv2BhVd8vWy7PBim1M4"}
```

To receive information about the DCX such as its description, location, and firmware version, you can use the System information request. For the DCX to accept the request you must provide the token that was received when you authenticated. Replace <TOKEN> in the request below with the token provided.

Request:

```
GET /api/system/systemInfo HTTP/1.1
Host: 192.168.1.22
Accept: application/json
Content-Type: application/json
Authorization: Bearer <TOKEN>
```

Response:

```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Cache-Control: max-age=3, must-revalidate
App-Version: 0.3.94
ETag: "b9-YuTkFdP1UR66Ii5k4OeZ4Q"
Access-Control-Expose-Headers: ETag
Content-Type: application/json; charset=utf-8
Content-Length: 340
Vary: Accept-Encoding
Date: Wed, 09 Nov 2016 15:59:08 GMT
Connection: keep-alive
```

{"description":"BlackBox DCX","location":"Unspecified","systemVersion":"3.00","firmwareVersion":"3.00.0272","recoveryVersion":"1.03.4323","boardVersion":2,"datafpgaVersion":"000b","videofpgaVersion":"00.0b","ignoreFirmwareMismatch":false,"imageType":"PRIMARY","edidId":101,"links":{"self":"/api/system/systemInfo"},"webVersion":"0.3.94"}

To connect a Receiver to a Transmitter, you use the consoles request in the following format…

/api/consoles/{id}/switch

{id} = The port number that the console is connected to.

In the body of the request, you specify the computers id. E.g. the port number it is connected to and the view mode which can be VIEWONLY, SHARED, EXCLUSIVE or PRIVATE. The response will be 204 "No Content" if it is successful.

Request:

```
POST /api/consoles/1/switch HTTP/1.1
Host: 10.0.10.14
Connection: keep-alive
Content-Length: 48
Content-Type: application/json
Accept: application/json
Authorization: Bearer <TOKEN>
{
"computerId": 1,
"mode": "VIEWONLY"
}
```

Response:

```
HTTP/1.1 204 No Content
X-Powered-By: Express
Access-Control-Allow-Origin: *
Cache-Control: max-age=3, must-revalidate
App-Version: 0.3.94
Date: Wed, 09 Nov 2016 15:46:15 GMT
Connection: keep-alive
```

More Information

There are many resources on the internet on how to use RESTful (Representational state transfer) API's/webservices.

Below is a list of useful resource links:
https://en.wikipedia.org/wiki/Representational_state_transfer
https://www.tutorialspoint.com/restful/restful_introduction.htm
http://restcookbook.com/