

# Toon Open API documentation

Version 1.2

## 1. Steps for credentials

- 1.1. Register on <http://www.toonapi.com/>. You will receive an email.
- 1.2. Verify your account by clicking the link in the received email. You will receive an email with the API Key, generated Password, Consumer-key and Consumer-secret.

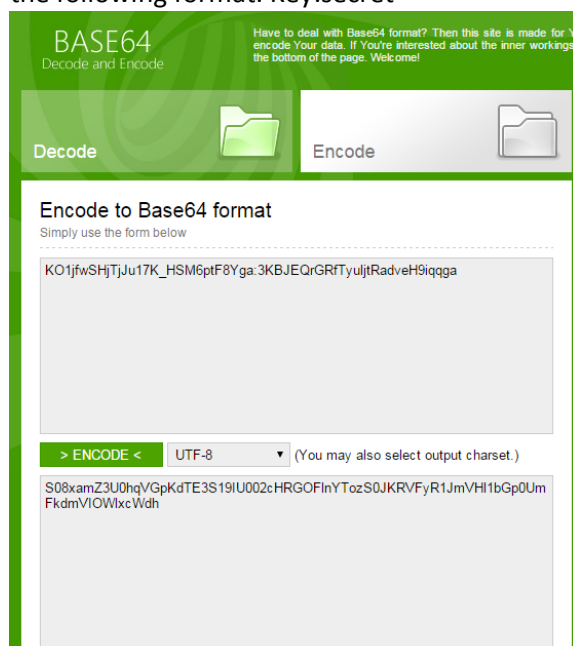
## 2. Steps for using Node.js app

- 2.1. Go to our Github page <https://github.com/quby-energy/ToonDemoApplication> and clone the repository or download the zip file.
- 2.2. Extract the zip file and edit the app.js.
- 2.3. Go to <https://nodejs.org/> and download and install Node.js.
- 2.4. Open a command line windows and navigate to or stay in the folder where you have your app.js file. Run the command `'npm install'`.
- 2.5. Edit line 17, 18, 19 and 20 to contain the account info you received in the e-mail.
- 2.6. Open a command line windows and navigate to or stay in the folder where you have your app.js file. Open the application in Node.js by entering the command `'node app.js'` or `'npm start'`.  
*The application uses port 8080 and 3001 so you need to make sure these ports are available. Also make sure your user has root/administrator rights.*

- 2.7. Now that the application is running, you can go to <http://localhost:8080> and press the Login button. You are now in the Toon API Simulator.

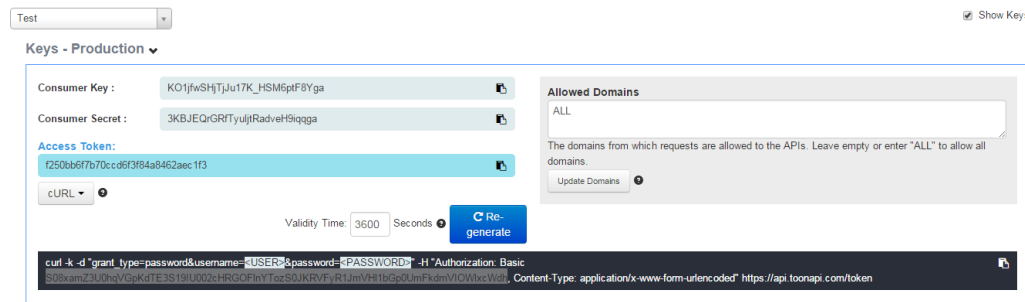
## 3. Steps for making REST calls to the API

- 3.1. Pick your weapon of choice for making REST calls. For instance cURL or Postman for Google Chrome.
- 3.2. Generate your authentication header in one of the two ways described below.
  - 3.2.1. Go to <https://www.base64encode.org/> and input Consumer-Key and Consumer-Secret in the following format. Key:secret



The screenshot shows the 'BASE64 Decode and Encode' website. It has a green header with the site name and a brief description. Below the header, there are two tabs: 'Decode' (active) and 'Encode'. The 'Encode' tab is selected, showing a form titled 'Encode to Base64 format'. The form has a text input field containing the text 'KO1jfwSHjTJu17K\_HSM6ptF8Yga:3KBJEQrGRfTyuljtRadveH9iqqga'. Below the input field, there is a button labeled '> ENCODE <' and a dropdown menu set to 'UTF-8'. To the right of the dropdown, it says '(You may also select output charset.)'. Below the button and dropdown, there is a text output field showing the encoded result: 'S08xamZ3U0hqVGpKdTE3S19IU002cHRGOFInYT0zS0JKRVFyR1JmVHl1bGp0UmFkdMvIOWlxcWdh'.

- 3.2.2. Go to <https://api.toonapi.com/store> and go to My Subscriptions. Select the pre-created ToonAPIApplication. Select the Curl command for GrantType: Password. Copy the Authentication Header in the curl command.



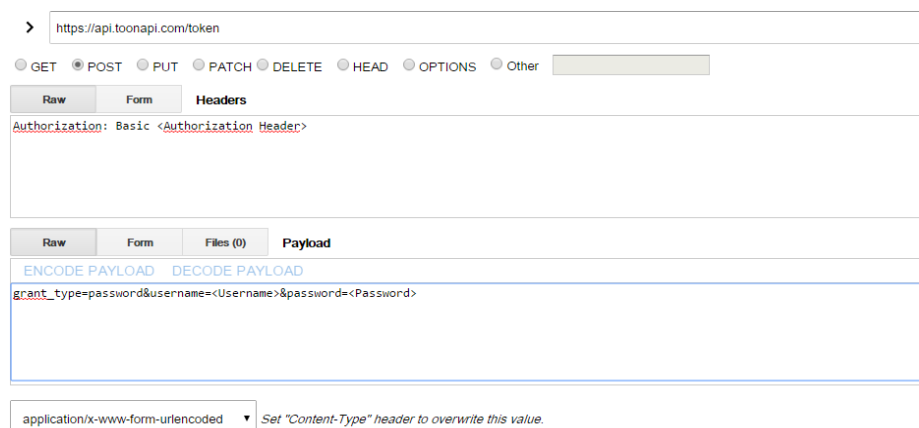
- 3.3. Make the call to the token endpoint to receive an access token. You can either import the whole cURL command from step 2b into a command line and replace the username and password field or create the call yourself.

- 3.4. The call comprises of a few elements. The url is <https://api.toonapi.com/token> and the method is POST.

Add two headers: Authorization: Basic **YOUR\_AUTHENTICATION\_HEADER**, Content-Type: application/x-www-form-urlencoded.

Add three elements to the body: grant\_type=password, username=**USERNAME**, password=**PASSWORD**.

You will receive a JSON object containing the access token.



*Example of the call in the Advaned Rest Client (Google Chrome Extension)*

- 3.5. All other calls to the API will have url <https://api.toonapi.com/toon/api/v1/OPERATION> and a header with the Access token. The next call you should execute is a GET call on /agreements. Add a header for Authorization: Bearer **ACCESS\_TOKEN**. You will receive an array of JSON objects with agreementIds. These agreements indicate to which Toon devices you have access.

- 3.6. You then need to set the Toon device to which you want to make API calls. For example, if you want to change or retrieve the temperature for a certain Toon device, you need to set its agreementId as the current.

To set the agreementId of a certain Toon device make a POST call on /agreements. Add two headers: the standard authentication Bearer header as described in step 3.5. and Content-Type: application/json. Then add a body with the agreement you want to use. The JSON object can be copied from the previous call or created manually. An example can be seen below.

```
{
  "agreementId": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
  "street": "xxxxxxxxxxxxxxxxxxxxxx",
  "houseNumber": "xxx",
  "postalCode": "xxxxxx",
  "city": "xxxxxxxxxx",
  "displayName": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",
  "displaySoftwareVersion": "xxxxxxx"
}
```

Only the agreementId is required, the other fields are optional. If the agreement is accepted it returns a 200 OK Http status code.

- 3.7. You can now check the status of that Toon device with a GET call on /status. Add the default authorization header. This returns a JSON object with information on the Toon device like current temperature and program. You can continue making calls to the API. Check the online console at <http://www.toonapi.com/documentation> to see the other possible calls.