



User Guide

Basic API Tutorials and Access

November 2022

Document Version 1.0



Table of Contents

| | |
|---|----------|
| PREREQUISITES..... | 3 |
| STEP 1: REGISTER FOR DEVELOPER KEYS | 3 |
| STEP 2: OBTAIN YOUR AUTHENTICATION KEYS | 6 |
| STEP 3: GETTING ACCESS TO THE API..... | 6 |
| OBTAIN YOUR ACCESS TOKEN..... | 6 |
| OTHER METHODS OF ACCESSING API CAN BE ACCOMPLISHED THROUGH AN APPLICATION CALLED POSTMAN. | 7 |
| OBTAIN YOUR PROFILE DETAILS..... | 7 |
| STEP 4: ACCESS A SINGLE RECORDING..... | 7 |
| OBTAIN THE ACCOUNT RECORDINGS | 7 |
| STEP 5: DOWNLOADING A RECORDING..... | 8 |
| OBTAINING RECORDING DATA FOR REPORTING..... | 9 |
| STEP 6: DOWNLOAD ALL YOUR RECORDINGS | 9 |



Prerequisites

You must have administration access to complete these steps

Step 1: Register for Developer Keys

To get your API developer keys you must first register through the Dubber Developer site.

1. Go to the Dubber Developer site and select register
<https://developer.dubber.net/member/register>
2. Fill out the information and press 'Register'

Register for an account

Register a new Mashery ID to access developer.dubber.net

Username

Display Name

This is the name which other users will see

Email

A validation E-mail will be sent to this address. Validation is required to complete registration.

Confirm E-mail

Please re-enter your e-mail address.

Password Requirements

- At least one letter
- At least one number
- At least eight characters

New Password

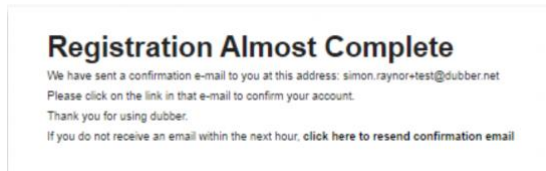
Passwords must be at least eight characters and contain at least one number and one letter.

Repeat New Password

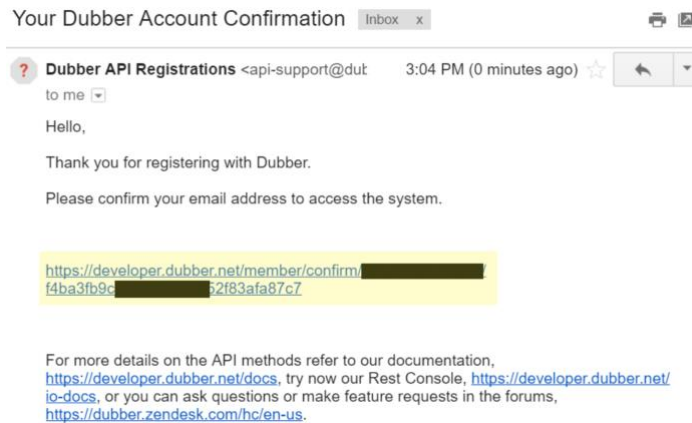
REGISTER

By clicking the "Register" button, I certify that I have read and agree to the Mashery Terms of Service and Privacy Policy

3. Check your email for a registration confirmation email



4. Click the confirmation link in the email



5. You now have registered successfully for a developer account. To get API keys you need to create an application and request production keys for environments relevant to your application. The Sandbox environment is automatically provided.

To do this click on the API link on the registration page

Registration Successful

You have successfully registered as [simon.raynor.test](#)

Next Steps

1. Apply for access to the **API**
2. Read the **API Documentation** for instructions on making calls.
3. Build something awesome.

Or if you have already left that page, go to the 'My Account' link at the top right of the page.

Signed in as [Test User](#) | [My Account](#) | [Sign Out](#)

6. Select the 'Get API Keys' link

My API Keys

[KEYS](#) | [APPLICATIONS](#) | [MANAGE ACCOUNT](#)

You don't have any keys yet!

[Get API Keys](#)

7. Fill out all the required information and make sure the tick box is checked for all the environments you plan to use (Additional locations can be added later by requesting new keys)



dubber Application Registration

Register Your New Application
Name of your application (you can change it later)

Web Site

Please describe what your application will do

Register Callback URL

Select which Web APIs this application will use
☒ Issue a new key for Dubber Sandbox Package

- ☒ Dubber Sandbox Plan
Playground sandbox environment

| Key Rate Limits | |
|-----------------|------------------|
| 2 | Calls per second |
| 5,000 | Calls per day |

☒ Issue a new key for Dubber UK Prod package
Production package from Dubber UK

- ☒ Dubber UK Prod Plan
Plan for Dubber UK Prod

| Key Rate Limits | |
|-----------------|------------------|
| 2 | Calls per second |
| 5,000 | Calls per day |

☒ Issue a new key for Dubber AU Prod package
Production package from Dubber AU

- ☒ Dubber AU Prod Plan

8. Start developing! Sandbox keys will already be there and active. Production regions will be activated by Dubber within 2 business days. Email notifications are sent when a status change occurs i.e. Waiting to Active.

If you are having trouble activating your keys, please reach out to support@dubber.net



Step 2: Obtain your authentication keys

1. Log into the Dubber portal
2. Click on the gear
3. Click on account
4. Click on API
5. Copy your Auth ID and Auth Token, these will be needed for later

Note: The Auth ID and Auth Token bound to the account you chose will still be restricted by its own privileges. Using administrator accounts will grant the largest access.

Step 3: Getting access to the API

Documentation for the API can be found here:

<https://developer.dubber.net/>

The interactive API can be leveraged to test your API calls without needing to do any programming.

<https://developer.dubber.net/io-docs>

To use the Interactive API page, you will require access to the sandbox. Your client ID and client secret are the mashery ID and mashery secret obtained in Step 1. This interactive page does not allow you to access your actual account but is useful for testing.

Obtain your access token

The API call we want to use is [Authorisation by Password](#)

If you access this token via a programming language, like python, you will copy the URL and post the values.

An example of this would be the following:



```
def getToken(mashery_key,mashery_secret, auth_id, auth_token):
    jump = "https://api.dubber.net/sandbox/v1/token?client_id={0}& \
    client_secret={1}&username={2}&password={3}& \
    grant_type=password".format(mashery_key,mashery_secret, auth_id, auth_token)
    r = requests.post(jump)
    print(f"Status Code: {r.status_code}, Response: {r.json()}")
    jsonResponse = r.json()
    access_token = jsonResponse['access_token']
    refresh_token = jsonResponse['refresh_token']
    return access_token,refresh_token
```

Be aware if you are not accessing the sandbox, you will need to change the URL to the correct environment.

Other methods of accessing API can be accomplished through an [application called Postman](#).

Obtain your profile details

Some of the additional commands the users emails and profile username. This information be obtained by gathering the profile details that are bound to the auth_id and auth_token. [Use the Get API resource profile call](#).

```
def getProfile(access_token):
    jump = "https://api.dubber.net/sandbox/v1/profile"
    headers = {"Authorization": "Bearer {}".format(access_token)}
    r = requests.get(jump,headers=headers)
    print(f"Status Code: {r.status_code}, Response: {r.json()}")
    jsonResponse = r.json()
    return jsonResponse
```

Step 4: Access a single recording

Obtain the account recordings

To access a single recording for downloading or meta data retrieval, the recording_id must be indicated. The following piece of python code will be able to gain a list of recent account recordings. In this example, the header must also now contain the authorisation token in bearer format. We pass the access_token gained earlier into the header for this call to be completed. Read [Get Account Recordings](#) for more details.



```
def getAccountRecordings(account_id, access_token):
    jump = "https://api.dubber.net/sandbox/v1/accounts/{1}/recordings" \
        .format(account_id)
    headers = {"Authorization": "Bearer {}".format(access_token)}
    r = requests.get(jump, headers=headers)
    print(f"Status Code: {r.status_code}, Response: {r.json()}")
    jsonResponse = r.json()
    recordings = jsonResponse['recordings']
    return recordings
```

Using the above command, we can take a single record and its recording_id and use it towards accessing a single record. Moreover, if the data is available, we can also retrieve the AI data for this record.

```
def getAIDetails(self, account_id, recording_id):
    jump = "https://api.dubber.net/sandbox/v1/accounts/{1}/recordings/{2}/ai" \
        .format(account_id, recording_id)
    headers = {"Authorization": "Bearer {}".format(self.access_token)}
    r = requests.get(jump, headers=headers)
    print(f"Status Code: {r.status_code}, Response: {r.json()}")
    jsonResponse = r.json()
    return jsonResponse
```

Step 5: Downloading a Recording

In this step we will down the recording, but we have limited time to download the recording from the URL for compliance reasons. There are two options we have with the API command, the first is to obtain the URL, and this is useful if we want to copy the recording from once space to another space. The other is to immediately collect the recording as an attachment. Depending on what you are trying to accomplish one method or the other may be preferable. Read more details on [Get Recording Link](#).

```
def getRecordingLink(recording_id, user_email, access_token):
    #url encoded
    #&content_disposition=attachment will provide the file directly \
    #otherwise it will return the URL
    #will need a save path
    u_email = urllib.parse.quote(user_email)
    jump="https://api.dubber.net/sandbox/v1/recordings/{0}?listener={1}" \
        .format(recording_id, u_email)
    headers = {"Authorization": "Bearer {}".format(access_token)}
    r = requests.get(jump, headers=headers)
    jsonResponse = r.json()
    print(f"Status Code: {r.status_code}, Response: {r.json()}")
    recording_url = jsonResponse['recording_url']
    return recording_url
```




Adding `&content_disposition=attachment` to the end of the url will immediately cause the recording to download.

Obtaining recording data for reporting

In this next step we will gather the recordings for the last 30 days and gain some additional data and AI data about the recordings themselves. We will leverage the data exporter function to extract this data. As part of this request, we must also provide a from-date in which 30 days will be counted from there. We can also provide a to_date to cap off the 30 days. For sharing the example code, I've hardcoded the dates with their time zone so showcase the example. Read more the [export function](#).

```
def getDataExport(account_id, start_date, from_date):  
    #'X-Date-Format: ISO8601'  
    dates = {"to_date": "2022-07-20T09:29:39+11:00", \  
            "from_date": "2022-06-20T09:29:39+10:00"}  
    jump="{0}/{1}/v1/accounts/{2}/data_exports".format(account_id)  
    headers = {"Authorization": "Bearer {}".format(access_token), \  
              "Content-Type": "application/json", "X-Date-Format": "ISO8601"}  
    r = requests.post(jump, headers=headers, json=dates)  
    jsonResponse = r.json()  
    print(f"Status Code: {r.status_code}, Response: {r.json()}")  
    return jsonResponse
```

Step 6: Download all your recordings

A common request is to download all your recordings for local long-term storage. This step is just a combination of the above steps, which is first we obtain an export of the recordings and using that export, we use the individual recording IDs to leverage the second function to download them all.



The output will appear like the following from the export

```
{'self': 'https://api.dubber.net/sandbox/v1/accounts/dubberdemo/recordings',
'recordings': [{'self': 'https://api.dubber.net/sandbox/v1/recordings/1175453523319619587',
'id': '1175453523319619587', 'to': 'QA Tester', 'to_label': 'QA Tester', \
'from': 'Zoom QA',
'call_type': 'inbound', 'recording_type': 'recording', 'channel': 'QA Tester',
'status': 'Active', 'start_time': 'Tue, 18 Oct 2022 05:12:15 +0100',
'duration': 5, 'dub_point_id': 'e0c639019f6511201f4c', 'tags': ['Test Call'],
'meta_tags': {'external-call-id': '1ecf7c1bb4c749f9bdfae196cb7a891f',
'original-filename': 'zoom-phone-Tlpxu61XQZyVpmdU3FCCNA-u6K0WU7oTrKW-63B4Ep4BA-1ecf7c1bb4c749f9bdfae196cb7a891f-2022-10-18T04:12:15Z.mp3',
'recorder-identifier': 'sbox.zoom', 'recording-platform': 'zoom'}, 'held': False, \
'type': 'audio', 'date_created': 'Tue, 18 Oct 2022 04:12:38 +0000',
'date_updated': 'Tue, 18 Oct 2022 04:12:42 +0000'},
{'self': 'https://api.dubber.net/sandbox/v1/recordings/1175448213674803203', \
'id': '1175448213674803203', 'to': '918618525652', 'from': 'QA Tester', 'from_label': 'QA Tester',
'call_type': 'outbound', 'recording_type': 'recording', 'channel': 'QA Tester', \
'status': 'Active', 'start_time': 'Tue, 18 Oct 2022 04:51:16 +0100', 'duration': 4,
'dub_point_id': 'e0c639019f6511201f4c', 'tags': ['Demo'], 'meta_tags': {'external-call-id': '5ee1f0924a7347238fae324d787d98d1', 'original-f
'recorder-identifier': 'sbox.zoom', 'recording-platform': 'zoom'}, 'held': False, 'type': 'audio', 'date_created': 'Tue, 18 Oct 2022 03:51:
'id': '1175448336744022019', 'to': 'meeting', 'from': 'qa+1@dubber.net', 'from_label': 'QA Tester', 'call_type': 'outbound', 'recording_typ
'dub_point_id': 'e0c639019f6511201f4c', 'meta_tags': {'external-call-id': 'LHPAQd/kRQC64TACj6LPzw==', 'original-filename': 'zoom-Tlpxu61XQZ
'recording-platform': 'zoom'}, 'held': False, 'type': 'video', 'date_created': 'Tue, 18 Oct 2022 03:52:02 +0000', 'date_updated': 'Tue, 18
```

You then send the individual ids into the recording link function to be able to download each one.



Need some help?

Browse through our 'How to' guides, tutorials, FAQ, and other resources - all designed to help and support teams to end not knowing.

support.dubber.net