

ExcelKey api manual

1, <https://rrpc.baasid.io:442/admin/loginForm>
join membership (Entering corporate information)

2. Admin Approval

3 Log in(Confirm issue key)

[4.http://api.baasid.com/](http://api.baasid.com/)

Entering and testing keys from the API menu

1. join membership



Secondary Authentication

User Information

[Sign Up](#)

Subscription Basic Information

Service Provider(OSP) Information			
+ Company Name	<input type="text"/>	CEO Name	<input type="text"/>
+ Company Registration Number	<input type="text"/>	+ Business location(Basic)	<input type="text"/>
	<input type="button" value="SearchInfo"/>		<input type="text"/>
Business(Sectors)	<input type="text" value="()"/>	+ Business location(Detail)	<input type="text"/>
+ Name of person in charge	<input type="text"/>	Business type	Service Provider <input type="button" value="v"/>
Contact phone number	<input type="text"/> - <input type="text"/> - <input type="text"/>	F A X	<input type="text"/> - <input type="text"/> - <input type="text"/>
+ Company phone number	<input type="text"/> - <input type="text"/> - <input type="text"/>		

User information	
+ User ID	<input type="text"/>
	<input type="button" value="Duplicate"/>
+ Name	<input type="text"/>
+ Password	<input type="text"/>
	6-10 characters including English and numbers
+ Confirm Password	<input type="text"/>
+ Mobile phone number	<input type="text"/> - <input type="text"/> - <input type="text"/>
User Level	Service Provider <input type="button" value="v"/>
+ E-MAIL	<input type="text"/> @ gmail.com <input type="button" value="v"/>

Terms of subscription

Secondary authentication management system (hereinafter referred to as management system) is a system that requests and can be issued by requesting the key for the authorization and the key for API call for the use of secondary authentication. And you have to pay for the contract.

The settlement is based on monthly payments, and you can check the expenses corresponding to the items agreed upon at the time of joining.

☐ I have read and agree to all the above terms and conditions.

Privacy Statement

The secondary authentication management system (hereinafter referred to as the management system) complies with the personal information protection regulations in related laws such as the Personal Information Protection Act. In the case of the homepage of the management system that processes personal information, Article 30 (1) of the Personal Information Protection Act and in accordance with Article 31 (1) of the Enforcement Decree, the Personal Information Processing Policy is established and operated on the relevant homepage. In case of changing the Personal Information Processing Policy, the information subject is compared before and after the change for easy confirmation. Will be made public.

☐ I have read and agree to all the above terms and conditions.

[Regist](#)[Cancel](#)

Description

1. Enter company information.
2. Click on the Regist

Description

Please let 'yds.baasid@gmail.com' know your registered ID after registration

Secondary Authentication

User Information

Sign Up

Subscription Basic Information

Service Provider(OSP) Information

* Company Name

* Company Registration Number

Search Info

Business(Sectors)

()

* Name of person in charge

Contact phone number

* Company phone number

CEO Name

* Business location(Basic)

* Business location(Detailed)

Business type

Service Provider

X

* User ID

Duplicate

* Password

6-10 characters including English and numbers

* Mobile phone number

* E-MAIL

@ gmail.com

* Name

* Confirm Password

User Level

Service Provider

Terms of subscription

2. Admin Approval

https://rrpc.baasid.io:442/admin/loginForm



Secondary Authentication

Description

1. Input User ID & Password
2. Click on the "Sign In" button



User ID

jykim

Password

.....

Sign In

ID/Password Find Sign Up

3. Log in (Confirm issue key)

https://rrpc.baasid.io:442/admin/loginForm



Secondary Authentication

Administration

[Dashboard](#) >[Authorization Key](#)[Statistics](#) >[Billing](#) >

Request to generate API authorization key for service provider

※ Please enter your service provider's information.

1

※ Please enter your user information to extend the key usage period.

1

Look up the user's key.

All



Number of results : 0

	Key	OSP_NAME	Start Date	End Date
No results were found for your search.				
Page 1 of 0				

Description

1. Input Unique Key.

"Unique Key" is the "App ID" that is a random value generated to distinguish between developers' apps
2. Click on the "Key generation request"

Description

1. Key issue completed

Administration

Dashboard

Authorization Key

Statistics

Billing

Request to generate API authorization key for service provider

※ Please enter your service provider's information.

1

Enter a unique key for the test device

Key generation request

※ Please enter your user information to extend the key usage period.

1

Enter a unique key for the test device

Request extension key

Look up the user's key.

All

Search

Number of results : 1

	Key	OSP_NAME	Start Date	End Date
1	8HKNDXeAjz2eC3+RIH/9pzTOI2Bp2TJ1IHtzBNqQqVooY9+1r1UoUyIDhDyHzBBD	BaaSidsss	2020-02-18 15:09:00	2021-02-18 15:09:00

Page 1 of 1 20

Verify

Your key issue request has been registered. The key that can be used for secondary authentication was sent via e-mail registered at the time of registration.

Close


PROJECT NAME	PAGE NAME	PAGE NUMBER	
<div><div>[2nd Authentication Key] Send the key via mail to the user. 받은편지함 ×</div><div><div>BaaS LAB Auth <baaswallet@gmail.com>나에게 ▾</div><div>오후 3:09 (2분 전) ☆ ↶ ⋮</div><div>🌐 영어 ▾ > 한국어 ▾ 메일 번역영어 번역 안함 ×</div><div>The transmitted key must be managed by the user and be careful not to be exposed outside The key can be used for one year and can be provided by renewal if the term expires. Secondary Authentication Key is <div>ASDF84jIn4K4nG/H5nT84G6G5J4G+6J8SFDG548SFGe45+8/R6FDS5G46SFG45A==S</div></div><div>↶ 답장 ➡ 전달</div></div></div>			<div>Description</div> <div>1. You can check the key price by email.</div>

**4. https://rrpc.baasid.io:442/v2/auth/auth_api_guide
Entering and testing keys from the API menu**

https://rrpc.baasid.io:442/v2/auth/auth_api_guide

Description

1. Click on "Authorize"



https://rrpc.baasid.io:442/websocket/websocket

Demonstration

Secondary Authentication 1.0.1


[Base URL: rrpc.baasid.io/v2]
https://rrpc.baasid.io:442/v2/auth/get_info


This is a test server Test-Net server. You can find out more about Secondary Authentication at <https://rrpc.baasid.io:442> or on [demonstration site, #eXcelDrive](#). For this site, you can use the authentication key **User-Authentication** to test the Request Header.


[Terms of service](#)
[Contact the developer](#)
[Apache 2.0](#)


Schemes


HTTPS


Authorize 


2nd Auth (Secondary Authentication) 


POST /auth/prepare_nego To start the 2nd auth. 

POST /auth/confirm_nego Provides the ability to register a service provider. 

POST /auth/register_device Provides a function to register a terminal identifier. 

POST /auth/request_auth Provides the ability to request secondary authentication. 

POST /auth/confirm_auth Provides a function for the user to perform secondary authentication verification. 



<https://rpc.baasid.io:442/websocket/websocket>

Demonstration

Secondary Authentication

1.0.1

[Base URL: <https://rpc.baasid.io:442>]
https://rpc.baasid.io:442/auth/get_info

This is a test server Test-Net server. You can find out more about Secondary Authentication at <https://rpc.baasid.io:442> or on [demonstration site](#). #ExcelDrive. For this site, you can use the authentication key `User-Authentication` to test the Request Header.

[Terms of service](#)
[Contact the developer](#)
Apache 2.0

Schemes

HTTPS

Authorize

Close

2nd Auth (Secondary Authentication)

POST

[/auth/prepare_nego](#)

To start the 2nd authentication.

POST

[/auth/confirm_nego](#)

Provides the ability to register a service provider.

POST

[/auth/register_device](#)

Provides a function to register a terminal identifier.

POST

[/auth/request_auth](#)

Provides the ability to request secondary authentication.


POST

[/auth/confirm_auth](#)

Provides a function for the user to perform secondary authentication verification.

Description

1. Enter the issued key

PROJECT NAME		PAGE NAME		PAGE NUMBER	16
<div><div></div><div><div>https://rrpc.baasid.io:442/websocket/websocket</div><div>Demonstration</div></div></div>					Description
<div><div><h1>Secondary Authentication</h1><div>1.0.1</div><div>[Base URL: rrpc.baasid.io/v2] https://rrpc.baasid.io:442/v2/auth/get_info</div><div><p>This is a test server Test-Net server. You can find out more about Secondary Authentication at https://rrpc.baasid.io:442 or on demonstration site, #eXcelDrive. For this site, you can use the authentication key User-Authentication to test the Request Header.</p><div>Terms of service Contact the developer Apache 2.0</div></div></div></div>					1. Test the 2 nd Auth API.
<div><div><div>Schemes<div>HTTPS</div></div><div>Authorize</div></div><div><div><h2>2nd Auth (Secondary Authentication)</h2><div><div>POST</div><div>/auth/prepare_nego</div><div>To start the 2nd auth.</div><div></div></div><div><div>POST</div><div>/auth/confirm_nego</div><div>Provides the ability to register a service provider.</div><div></div></div><div><div>POST</div><div>/auth/register_device</div><div>Provides a function to register a terminal identifier.</div><div></div></div><div><div>POST</div><div>/auth/request_auth</div><div>Provides the ability to request secondary authentication.</div><div></div></div><div><div>POST</div><div>/auth/confirm_auth</div><div>Provides a function for the user to perform secondary authentication verification.</div><div></div></div></div></div></div>					

1) /auth/prepare_nego

- Description
1. Enter parameters

2. Click on "Try it out"

3. Click on "Execute"

2nd Auth (Secondary Authentication)

POST

/auth/prepare_nego

To start the 2nd auth.

Returns none

Parameters

Try it out

Name	Description
<div>requestor</div> <div><div>*</div> required</div> <div>string</div> <div>(request_body)</div>	<div>Service provider or OSP's name</div> <div>BaaSlab</div>
<div>ticketNo</div> <div><div>*</div> required</div> <div>string</div> <div>(request_body)</div>	<div>Number generated by the server and given to the user.</div> <div>afeuhf342aldskjf65612351</div>

Execute

Responses

Response content type

application/json

PROJECT NAME	PAGE NAME	PAGE NUMBER	18						
			Description						
<div>1) /auth/prepare_nego</div> <div><div>Execute</div><div>Clear</div></div> <div>Responses<div>Response content typeapplication/json</div></div> <div>Curl<pre>curl -X POST "http://rrpc.basaid.io:442/v2/auth/prepare_nego" -H "Content-Type: application/json" -H "User-Authorization: zfkdw8nZydlz088fomtA4Lh00qFtPov4Nye8laecx/nTg56DWMKA01MauI-H767o/p5l0cptyZp1IMORDPu0UA—" -d '{"requestor": "Bas8lab", "ticketNo": "afheulaa1f18861df"}'</pre></div> <div>Request URL<pre>https://rrpc.basaid.io:442/v2/auth/prepare_nego</pre></div> <div>Server response<table><tr><th>Code</th><th>Details</th></tr><tr><td>200</td><td><div>Response body<pre>{ "code": "200", "message": "The request entered was successfully processed.", "params": { "requestor": "Bas8lab", "ticketNo": "afheulaa1f18861df" } }</pre><div>Download</div></div><div>Response headers<pre>Content-type : application/json</pre></div></td></tr></table></div> <div>Responses<table><tr><th>Code</th><th>Description</th></tr></table></div>				Code	Details	200	<div>Response body<pre>{ "code": "200", "message": "The request entered was successfully processed.", "params": { "requestor": "Bas8lab", "ticketNo": "afheulaa1f18861df" } }</pre><div>Download</div></div> <div>Response headers<pre>Content-type : application/json</pre></div>	Code	Description
Code	Details								
200	<div>Response body<pre>{ "code": "200", "message": "The request entered was successfully processed.", "params": { "requestor": "Bas8lab", "ticketNo": "afheulaa1f18861df" } }</pre><div>Download</div></div> <div>Response headers<pre>Content-type : application/json</pre></div>								
Code	Description								

2) /auth/confirm_nego

- Description
1. Enter parameters

2. Click on "Try it out"

3. Click on "Execute"

POST

/auth/confirm_nego

Provides the ability to register a service provider.

Parameters

Try it out

Name	Description
<div>account <small>required</small> *</div> <div>string</div> <div>(request_body)</div>	<div>This parameter is the ID for the Wallet user.</div> <div>WalletID</div>
<div>requestor <small>required</small> *</div> <div>string</div> <div>(request_body)</div>	<div>Service provider or OSP's name</div> <div>BaaSlab</div>
<div>ticketNo <small>required</small> *</div> <div>string</div> <div>(request_body)</div>	<div>Number generated by the server and given to the user.</div> <div>asdf186a54dsfd ×</div>

Execute

Responses

Response content type

application/json

2) /auth/confirm_nego

Curl

```
curl -X POST https://rpc.basid.io:442/v2/auth/confirm_nego -H "Content-type: application/json" -H "User-Authorization: zhdw9n2ydlz069RontA4LhD0qFtPovt4Nye8Iaecx/nTg6ROMuKA01Nwul4R767o/p6I0cp4yZp1IMORDPu0UA—" -d '{"account": "WalletID", "requestor": "Bas8lab", "ticketNo": "asdf188a54defdfd"}'
```

Request URL

https://rpc.basid.io:442/v2/auth/confirm_nego

Server response

Code	Details
------	---------

200

Response body

```
{"code": "200", "message": "The request entered was successfully processed.", "params": [{"account": "WalletID", "requestor": "Bas8lab", "ticketNo": "asdf188a54defdfd"}]}
```

Download

Response headers

Content-type : application/json

Responses

Code	Description
------	-------------

3) /auth/register_device

Description

1. Enter parameters
2. Click on "Try it out"
3. Click on "Execute"

POST /auth/register_device Provides a function to register a terminal identifier.

Parameters

Try it out

Name Description

account required
*

This parameter is the ID for the Wallet user.

string
(request_body)

WalletID

deviceId required
*

Device that can receive the push service used by the secondary authentication service for verification.

string
(request_body)

deviceId

platform required
*

It refers to the operating system of a device that can use a push service that allows a user to verify authentication. It supports two types of Android and IOS.

string
(request_body)

Android

pushKey required
*

This parameter is the unique key value of the device for the push service, and this value can be changed by initialization or reset.

string
(request_body)

PushKey

x

Execute

Responses

Response content type

application/json



Description

3) /auth/register_device

Curl

```
curl -X POST "https://rpc.basaid.io:442/v2/auth/register_device" -H "Content-type: application/json" -H "User-Authorization: zHddW8nZydIzD89fKmtAALhDQqFtPovHMya8Iaecx/nTg6H0mKAOtHwUfR767o/p6l8cpHy2pIIMORFPuQUA—" -d '{"account": "WalletID", "deviceId": "deviceId", "platform": "Android", "pushKey": "PushKey"}'
```

Request URL

```
https://rpc.basaid.io:442/v2/auth/register_device
```

Server response

Code

Details

200

Response body

```
{"code": "200", "message": "The request entered was successfully processed.", "param": [{"account": "WalletID", "deviceId": "deviceId", "platform": "Android", "pushKey": "PushKey"}]}
```

Download

Response headers

```
Content-type : application/json
```

Responses

4) /auth/request_auth

POST /auth/request_auth Provides the ability to request secondary authentication.

Parameters Try it out

Name	Description
requestor * <small>required</small> string <small>(request_body)</small>	Service provider or OSP's name <input type="text" value="BaaSslab"/>
token * <small>required</small> string <small>(request_body)</small>	This parameter refers to the function of performing authentication request through push service. <input type="text" value="afea68468haofjkaasfeafh3930"/>

Execute

Responses Response content type application/json

Description

1. Enter parameters
2. Click on "Try it out"
3. Click on "Execute"

5) /auth/confirm_auth

- Description
1. Enter parameters

2. Click on "Try it out"

3. Click on "Execute"

POST

/auth/confirm_auth

Provides a function for the user to perform secondary authentication verification.

Parameters

Try it out

Name	Description
<div>requestor</div> <div><div>★ required</div></div> <div>string</div> <div>(request_body)</div>	<div>Service provider or OSP's name</div> <div>BaaSlab</div>
<div>account</div> <div><div>★ required</div></div> <div>string</div> <div>(request_body)</div>	<div>This parameter is the ID for the Wallet user.</div> <div>Walletid</div>
<div>message</div> <div><div>★ required</div></div> <div>string</div> <div>(request_body)</div>	<div>This parameter refers to the function to receive a message that is exposed to the user's app or the web after authentication is completed.</div> <div>authentication is completed</div> <div>×</div>

Execute

Responses

Response content type

application/json

5) /auth/confirm_auth

Curl

```
curl -X POST "http://rrpc.basaid.io:442/v2/auth/confirm_auth" -H "Content-Type: application/json" -H "User-Authorization: basaid" -d '{"account": "WalletId", "requestor": "Basaid", "message": "authentication is completed"}'
```

Request URL

```
http://rrpc.basaid.io:442/v2/auth/confirm_auth
```

Server response

Code

Details

200

Response body

```
{"code": "200", "message": "The request entered was successfully processed.", "params": [{"account": "WalletId", "requestor": "Basaid", "message": "authentication is completed"}]}
```

Download

Response headers

```
Content-type : application/json
```

Responses