

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/>
Entering and testing keys from the API menu**

1. join membership

Description

1. Click on the "Sign Up"

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

Login MyPage Feb 18, 2020 12:06:44



Secondary Authentication



Please enter your username and password.

User ID

Password

Sign In

ID/Password Find [Sign Up](#)



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"/> <input type="button" value="Search Info"/>	• Business location(Basic)	<input type="text"/>
Business(Sectors)	<input type="text"/> ()	• Business location(Detail)	<input type="text"/>
• Name of person in charge	<input type="text"/>	Business type	<input type="text" value="Service Provider"/>
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"/> <small>6-10 characters including English and numbers</small>	• Confirm Password	<input type="text"/>
• Mobile phone number	<input type="text"/> - <input type="text"/> - <input type="text"/>	User Level	<input type="text" value="Service Provider"/>
• E-MAIL	<input type="text"/> @ <input type="text" value="gmail.com"/>		

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 certification 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.

Description

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

Description

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"/> <input type="button" value="SearchInfo"/>	* Business location(Basic)	<input type="text"/>
Business(Sectors)	()	* Business location(Detailed)	<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"/>	* X	<input type="text"/> - <input type="text"/> - <input type="text"/>
* Company phone number	<input type="text"/>	on	
* User ID	<input type="text"/> <input type="button" value="Duplicate"/>	* Name	<input type="text"/>
* Password	<input type="text"/> <small>6-10 characters including English and numbers</small>	* 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

Confirm

User registration is complete. Go to the login screen.

2. Admin Approval

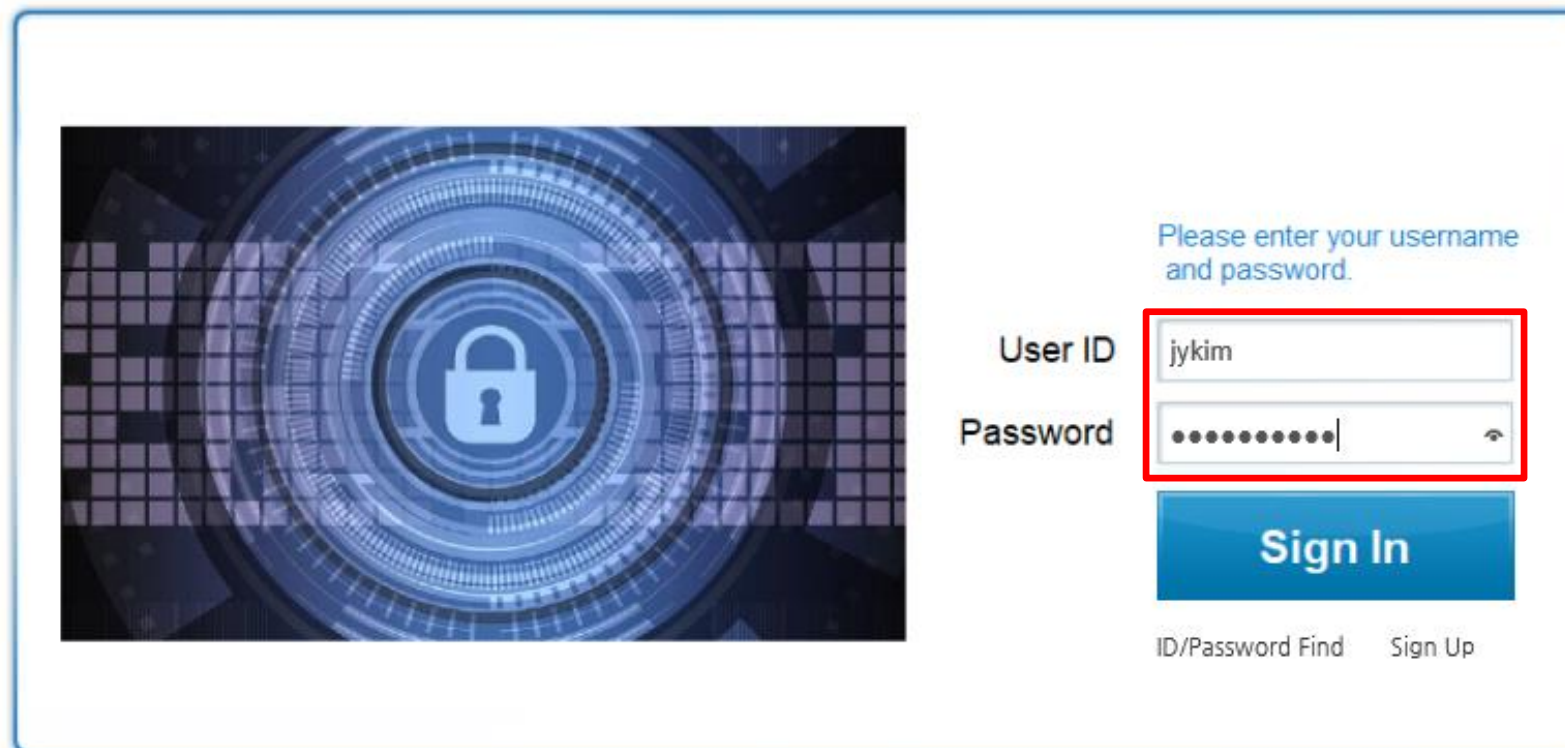
https://rrpc.baasid.io:442/admin/loginForm



Secondary Authentication

Description

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



The screenshot shows a login interface with a blue-themed background. On the left is a large graphic of a padlock surrounded by concentric circles and a grid. On the right, there is a text prompt: "Please enter your username and password." Below this are two input fields: "User ID" containing the text "jykim" and "Password" containing a series of dots. A red rectangular box highlights both input fields. Below the fields is a blue "Sign In" button. At the bottom of the form area, there are two links: "ID/Password Find" and "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

<input type="checkbox"/>	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

[kevindose / Kevin Dose] Logout MyPage Feb 18, 2020 15:10:23

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 Enter a unique key for the test device

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

1 Enter a unique key for the test device

Verify [X]

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.

Look up the user's key.

All

Number of results : 1

	<input type="checkbox"/>	Key	OSP_NAME	Start Date	End Date
1	<input type="checkbox"/>	8HKNdXeAjz2eC3+RIH/9pzTOI2Bp2TJ1IHfzBNqQqVooY9+1r1UoUyidDhDyHzBBD	BaaSidsss	2020-02-18 15:09:00	2021-02-18 15:09:00

Page 1 of 1 20

Description

1. You can check the key price by email.

[2nd Authentication Key] Send the key via mail to the user. 받은편지함 x



BaaS LAB Auth <baaswallet@gmail.com>

오후 3:09 (2분 전) ☆ ↶ ⋮

나에게 ▾

영어 ▾ > 한국어 ▾ 메일 번역

영어 번역 안함 x

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 **ASDF84jih4K4hG/H5hT84G6G5j4G+6j8SFDG548SFG45+8/R6FDS5G46SFG45A==**

↶ 답장

➡ 전달

**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-Authent i cat i on` to test the Request Header.

[Terms of service](#)

[Contact the developer](#)

[Apache 2.0](#)

Schemes

HTTPS

Authorize



2nd Auth (Secondary Authentication) ▼

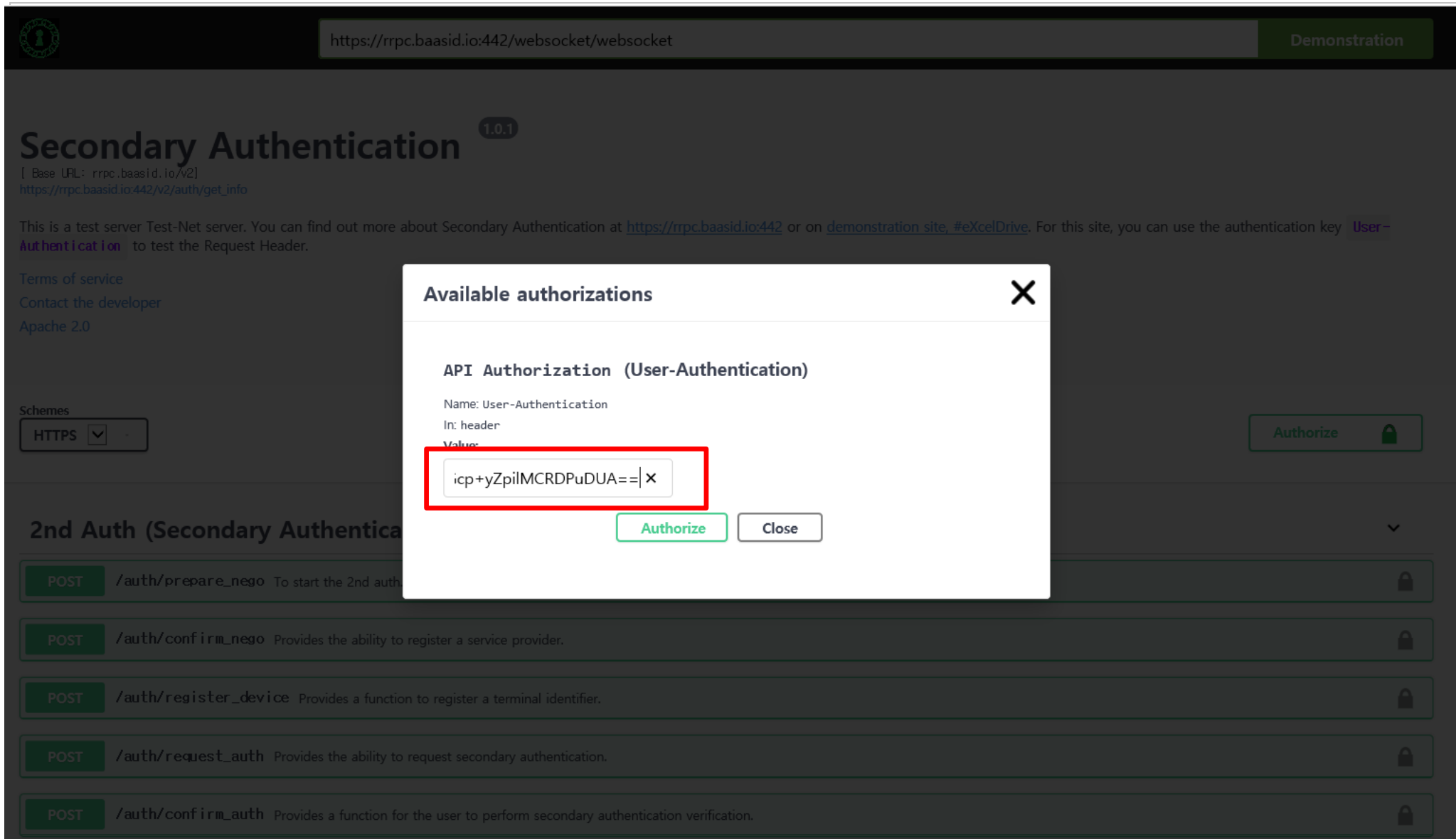
POST /auth/prepare_nego To start the 2nd auth.

POST /auth/conf i r m_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/conf i r m_auth Provides a function for the user to perform secondary authentication verification.



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

Available authorizations

API Authorization (User-Authentication)

Name: User-Authentication
In: header
Value:

Authorize Close

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

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

Demonstration

Description

1. Test the 2nd Auth API.

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-Authent i cat i on` 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/conf i r m_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/conf i r m_auth Provides a function for the user to perform secondary authentication verification.

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
------	-------------

requestor * required	Service provider or OSP's name
--------------------------------	--------------------------------

string (request_body)	<input type="text" value="BaaSlab"/>
--------------------------	--------------------------------------

ticketNo * required	Number generated by the server and given to the user.
-------------------------------	---

string (request_body)	<input type="text" value="afeuhf342aldskjf65612351"/>
--------------------------	---

Execute

Responses

Response content type

application/json

Description

1) /auth/prepare_nego

Execute

Clear

Responses

Response content type

application/json

Curl

```
curl -X POST "http://rpc.baaaid.io:442/v2/auth/prepare_nego" -H "Content-Type: application/json" -H "User-Authorization: zhdw8n2ydlz068RkntA4LH00qFtPovHny8Iaecx/nTp6ROW#KA0t1MuI-H767o/p6I0cp+yZpIIMORDPUDUA—" -d '{"requestor": "Baa8lab", "ticketNo": "aTheuIasI16861d1"}'
```

Request URL

```
http://rpc.baaaid.io:442/v2/auth/prepare_nego
```

Server response

Code

Details

200

Response body

```
{"code": "200", "message": "The request entered was successfully processed.", "params": {"requestor": "Baa8lab", "ticketNo": "aTheuIasI16861d1"}}
```

Download

Response headers

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

Responses


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

Execute

Responses Response content type

Description

2) /auth/confirm_nego

Curl

```
curl -X POST https://rrpc.baaaid.io:442/v2/auth/confirm_nego -H "Content-type: application/json" -H "User-Authzation: zhdw6nZydIz068RkntA4Lh00qFtPovHMya8Iaecx/nTg6R0MUKA0tNAuI+r767o/p6I6cp+yZpIIMDRPUDUA—" -d '{"account": "WalletID", "requestor": "Baa8lab", "ticketNo": "aadf188a54dafdfd"}'
```

Request URL

```
https://rrpc.baaaid.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": "Baa8lab", "ticketNo": "aadf188a54dafdfd"}}
```

[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 <small>required</small> *	This parameter is the ID for the Wallet user.
string <small>(request_body)</small>	WalletID
deviceID <small>required</small> *	Device that can receive the push service used by the secondary authentication service for verification.
string <small>(request_body)</small>	deviceID
platform <small>required</small> *	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 <small>(request_body)</small>	Android
pushKey <small>required</small> *	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 <small>(request_body)</small>	PushKey

Execute

Responses

Response content type

application/json

Description

3) /auth/register_device

Curl

```
curl -X POST https://rrpc.baaaid.io:442/v2/auth/register_device -H "Content-Type: application/json" -H "User-Authorization: zhdw8nZydlz068RkrtA4LhD0qFtPovHys8Iaecx/nTgGROWKAO1NwU1H767o/p616cptYzplIMORDPuQUA—" -d '{"account": "WalletID", "deviceID": "deviceID", "platform": "Android", "pushKey": "PushKey"}'
```

Request URL

```
https://rrpc.baaaid.io:442/v2/auth/register_device
```

Server response

Code

Details

200

Response body

```
{"code": "200", "message": "The request entered was successfully processed.", "params": {"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> <small>string</small> <small>(request_body)</small>	Service provider or OSP's name <input type="text" value="BaaSlab"/>
token <small>required *</small> <small>string</small> <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"

4) /auth/request_auth

Curl

```
curl -d '{"requestor": "BaaS LAB", "token": "eyJ0eXAiOiJKV1QiOiJhbnQ1IiwiaWF0IjoiYjZmYzIjLCJmcm9udCI6ImVudCJ9"}' https://trpc.baaaid.io:442/v2/auth/request_auth
```

Request URL

```
https://trpc.baaaid.io:442/v2/auth/request_auth
```

Server response

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


200	successful operation
-----	----------------------

Example Value | Model

```
{
  "code": "200",
  "message": "The request entered was successfully processed.",
  "params": {
    "requestor": "BaaS LAB",
    "token": "eyJ0eXAiOiJKV1QiOiJhbnQ1IiwiaWF0IjoiYjZmYzIjLCJmcm9udCI6ImVudCJ9"}
}
```

Description

5) /auth/confirm_auth

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

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="BaaSlab"/>
account * <small>required</small> string <small>(request_body)</small>	This parameter is the ID for the Wallet user. <input type="text" value="Walletid"/>
message <small>* required</small> string <small>(request_body)</small>	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. <input type="text" value="authentication is completed"/>

[Execute](#)

Responses Response content type

Description

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

Description

5) /auth/confirm_auth

Curl

```
curl -X POST "https://rrpc.bas81d.io:442/v2/auth/confirm_auth" -H "Content-Type: application/json" -H "User-Authorization: bas81ab" -d '{"account": "WalletId", "requestor": "Bas81ab", "message": "authentication is completed"}'
```

Request URL

```
https://rrpc.bas81d.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": "Bas81ab", "message": "authentication is completed"}}
```

[Download](#)

Response headers

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

Responses