

EIYARO Core API Reference

Table of Contents

Wallet Endpoints	3
Create Key Endpoint	4
Parameters	4
Returns	4
Example	4
List Keys Endpoint	5
Parameters	5
Returns	5
Example	5
Update Key Alias Endpoint	6
Parameters	6
Returns	6
Example	6
Delete Key Endpoint	7
Parameters	7
Returns	7
Example	7
Check Key Password Endpoint	7
Parameters	7
Returns	8
Example	8
Reset Key Password Endpoint	8
Parameters	8
Returns	8
Example	8
Create Account Endpoint	9
Parameters	9
Returns	9
Example	10
List Accounts Endpoint	10
Parameters	10
Returns	10
Example	11
Update Account Alias Endpoint	11
Parameters	11
Returns	11

Example	11
Delete Account Endpoint	12
Parameters	12
Returns	12
Example	12
Create Account Receiver Endpoint	13
Parameters	13
Returns	13
Example	13
List Addresses Endpoint	13
Parameters	14
Returns	14
Example	14
Validate Address Endpoint	15
Parameters	15
Returns	15
Example	15
Get Mining Address Endpoint	15
Parameters	15
Returns	16
Example	16
Set Mining Address Endpoint	16
Parameters	16
Example	16
Get Coinbase Arbitrary Endpoint	17
Parameters	17
Returns	17
Example	17
Set Coinbase Arbitrary Endpoint	17
Parameters	17
Returns	18
Example	18
List pubkeys Endpoint	18
Parameters	18
Returns	18
Example	19
Create Asset Endpoint	19
Parameters	19
Returns	20
Example	20
Get Asset Endpoint	21

Parameters	22
Returns	22
Example	22
List Assets Endpoint	23
Parameters	23
Returns	23
Example	23
Update Asset Alias Endpoint	24
Parameters	24
Returns	25
Example	25
List Balances Endpoint	25
Parameters	25
Returns	25
Example	25
List Unspent Outputs Endpoint	26
Parameters	27
Returns	27
Example	27
Backup Wallet Endpoint	29
Parameters	29
Returns	29
Example	29
Restore Wallet Endpoint	31
Parameters	31
Returns	31
Example	31
Rescan Wallet Endpoint	32
Parameters	32
Returns	32
Example	32
Network Endpoints	33

Wallet Endpoints

These endpoints are available when we set:
config.toml

```
[wallet]
disable = false
```

This is the default value and we can possibly omit it.

Create Key Endpoint

Creates a private key. The private key is encrypted in the file and not visible to the user.

Parameters

Object:

- **String** - **alias**, name of the key.
- **String** - **password**, password of the key.
- **String** - **language**, mnemonic language of the key.

Optional:

- **String** - **mnemonic**, mnemonic of the key, create key by specified mnemonic.

Returns

Object:

- **String** - **alias**, name of the key.
- **String** - **xpub**, root pubkey of the key.
- **String** - **file**, path to the file of key.

Optional:

- **String** - **mnemonic**, mnemonic of the key, exist when the request mnemonic is null.

Example

Create key by random pattern:

Request

```
curl -X POST http://localhost:9888/create-key -d '{"alias": "alice", "password": "123456", "language": "en"}'
```

Response

```
{
  "alias": "alice",
  "xpub":
"a85e6eccb22f4c5fdade905f9a969003a17b6f35c237183a4313354b819a92689d52da3bcfe55f15a5508
77e8d789bd2bb9620f46e5049ea36470ab1b588a986",
  "file": "/home/yang/.eiyaro/keystore/UTC--2024-3-10T07-09-17.509894697Z--341695b9-
```

```
9223-470c-a26d-bea210f8e1bb",
  "mnemonic": "verb smoke glory dentist annual peanut oval dragon fiction current
orbit lab load total language female mushroom coyote regular toy slide welcome employ
three"
}
```

Create key by specified mnemonic:

Request

```
curl -X POST http://localhost:9888/create-key -d '{"alias":"jack",
"password":"123456", "mnemonic":"please observe raw beauty blue sea believe then boat
float beyond position", "language":"en"}'
```

Response

```
{
  "alias": "jack",
  "xpub":
"c7bcb65febd31c6d900bc84c386d95c3d5b047090628d9bf5c51a848945b6986e99ff70388018a7681fa3
7a240dbd8df39a994c86f9314a61e75feb33563ca72",
  "file": "/home/yang/.eiyaro/keystore/UTC--2024-3-10T07-08-51.815030323Z--46ee932e-
88d3-4680-a5c1-dd9e63918fcc"
}
```

List Keys Endpoint

Returns the list of all available keys.

Parameters

None.

Returns

- **Array of Object**, keys owned by the client.
 - **Object**:
 - **String** - **alias**, name of the key.
 - **String** - **xpub**, pubkey of the key.

Example

Request a list of the current keys on the node.

Request

```
curl -X POST http://localhost:9888/list-keys
```

Response

```
[
  {
    "alias": "alice",
    "xpub":
"a7dae957c2d35b42efe7e6871cf5a75ebd2a0d0e51caff767db42d3e6d69dbe211d1ca492ecf05908fe6
fa625ad61b3253375ea744c9442dd5551613ba50aea",
    "file": "/Path/To/Library/Eiyaro/keystore/UTC--2024-03-21T02-35-15.035935116Z--
4f2b8bd7-0576-4b82-8941-6cc6da05efe3"
  },
  {
    "alias": "bob",
    "xpub":
"d30a810e88532f73816b7b5007d413cbd21e526ae9159023e5262511893adc1526b8eacd691b27c080201
d7d79336a4f3d2cb4c167d997821cad445765916254",
    "file": "/Path/To/Library/Eiyaro/keystore/UTC--2018-03-22T06-30-27.609315219Z--
0e34293c-8856-4f5f-b934-37456a3820fa"
  }
]
```

Update Key Alias Endpoint

Update the alias for an existing key.

Parameters

Object:

- **String** - **xpub**, pubkey of the key.
- **String** - **new_alias**, new alias of the key.

Returns

Nothing in case the key alias is updated successfully.

Example

Update an existing key's alias.

Request

```
curl -X POST http://localhost:9888/update-key-alias -d '{"xpub":  
"a7dae957c2d35b42efe7e6871cf5a75ebd2a0d0e51caff767db42d3e6d69dbe211d1ca492ecf05908fe6  
fa625ad61b3253375ea744c9442dd5551613ba50aea", "new_alias": "new_key"}'
```

Response

No response in case operation was successful.

Delete Key Endpoint

Deletes an existing key.



Please make sure that there is no balance in the related accounts.

Parameters

Object:

- **String** - **xpub**, pubkey of the key.
- **String** - **password**, password of the key.

Returns

Nothing in case the key is deleted successfully.

Example

Delete an existing key.

Request

```
curl -X POST {bas-url}delete-key -d '{"xpub":  
"a7dae957c2d35b42efe7e6871cf5a75ebd2a0d0e51caff767db42d3e6d69dbe211d1ca492ecf05908fe6  
fa625ad61b3253375ea744c9442dd5551613ba50aea", "password": "123456"}'
```

Response

No response in case operation was successful.

Check Key Password Endpoint

Check an existing key's password.

Parameters

Object:

- **String** - **xpub**, pubkey of the key.
- **String** - **password**, password of the key.

Returns

Object:

- **Boolean** - **check_result**, if check is successful the value will be true, otherwise it will be false.

Example

Check the password for an existing key.

Request

```
curl -X POST http://localhost:9888/check-key-password -d '{"xpub":  
"a7dae957c2d35b42efe7e6871cf5a75ebd2a0d0e51caffe767db42d3e6d69dbe211d1ca492ecf05908fe6  
fa625ad61b3253375ea744c9442dd5551613ba50aea", "password": "123456"}'
```

Response

```
{  
  "check_result": true  
}
```

Reset Key Password Endpoint

Reset an existing key's password.

Parameters

Object:

- **String** - **xpub**, pubkey of the key.
- **String** - **old_password**, old password of the key.
- **String** - **new_password**, new password of the key.

Returns

Object:

- **Boolean** - **changed**, if reset is successful the value will be true, otherwise it will be false.

Example

Reset the password for an existing key.

Request

```
curl -X POST http://localhost:9888/reset-key-password -d '{"xpub":  
"a7dae957c2d35b42efe7e6871cf5a75ebd2a0d0e51caff767db42d3e6d69dbe211d1ca492ecf05908fe6  
fa625ad61b3253375ea744c9442dd5551613ba50aea", "old_password": "123456",  
"new_password": "654321"}'
```

Response

```
{  
  "changed": true  
}
```

Create Account Endpoint

Create an account to manage addresses.

Single sign account contains only one **root_xpubs** and quorum; however multi sign account can contain any number of **root_xpubs** and quorum.

Quorum is the number of verify signatures, the range is `[1, len(root_xpubs)]`.

Parameters

Object:

- **Array of String** - **root_xpubs**, pubkey array.
- **String** - **alias**, name of the account.
- **Integer** - **quorum**, the default value is **1**, threshold of keys that must sign a transaction to spend asset units controlled by the account.

Optional:

- **String** - **access_token**, if optional when creating account locally. However, if you want to create account remotely, it's indispensable.

Returns

Object:

- **String** - **id**, account id.
- **String** - **alias**, name of account.
- **Integer** - **key_index**, key index of account.
- **Integer** - **quorum**, threshold of keys that must sign a transaction to spend asset units controlled by the account.
- **Array of Object** - **xpubs**, pubkey array.

Example

Create an account with a given `root_xpubs` and `alias`.

Request

```
curl -X POST http://localhost:9888/create-account -d
'{"root_xpubs":["2d6c07cb1ff7800b0793e300cd62b6ec5c0943d308799427615be451ef09c0304bee5
dd492c6b13aaa854d303dc4f1dcb229f9578786e19c52d860803efa3b9a"],"quorum":1,"alias":"alic
e"}'
```

Response

```
{
  "alias": "alice",
  "id": "08F0663C00A02",
  "key_index": 1,
  "quorum": 1,
  "xpubs": [

    "2d6c07cb1ff7800b0793e300cd62b6ec5c0943d308799427615be451ef09c0304bee5dd492c6b13aaa854
    d303dc4f1dcb229f9578786e19c52d860803efa3b9a"
  ]
}
```

List Accounts Endpoint

Returns a list of the available accounts on the node.

Parameters

Optional:

- `String` - `id`, account id.
- `String` - `alias`, name of account.

Returns

- `Array of Object`, account array.
 - `Object`:
 - `String` - `id`, account id.
 - `String` - `alias`, name of account.
 - `Integer` - `key_index`, key index of account.
 - `Integer` - `quorum`, threshold of keys that must sign a transaction to spend asset units controlled by the account.

- **Array of Object** - **xpubs**, pubkey array.

Example

Request a list of the accounts present on the node.

Request

```
curl -X POST http://localhost:9888/list-accounts -d '{"alias":"alice"}
```

Response

```
[
  {
    "alias": "alice",
    "id": "086KQD75G0A02",
    "key_index": 1,
    "quorum": 1,
    "xpubs": [
      "180aab8bf247932a7cf68da5cc9a873266279155097612f1e5fdda4add88d5e91e2e7ce5b736f3ac933824cdee9effcf1531b90dfcb388e5cc306d14e9a2c85e"
    ]
  }
]
```

Update Account Alias Endpoint

Updates an alias for the an existing account.

Parameters

Object: **account_alias** | **account_id**

* **String** - **new_alias**, new alias of account.

optional:

- **String** - **account_alias**, alias of account.
- **String** - **account_id**, id of account.

Returns

Nothing in case the account alias is updated successfully.

Example

Update the alias for a given account ID or an account alias.

Request

```
curl -X POST http://localhost:9888/update-account-alias -d '{"account_id":  
"08F0663C00A02", "new_alias": "new_account"}'  
# or  
curl -X POST http://localhost:9888/update-account-alias -d '{"account_alias": "alice",  
"new_alias": "new_account"}'
```

Response

No response in case operation was successful.

Delete Account Endpoint

Delete an existing account.



Please make sure that there is no balance in the related accounts.

Parameters

Object: `account_alias` | `account_id`

Optional:

- **String** - `account_alias`, alias of account.
- **String** - `account_id`, id of account.

Returns

Nothing if the account is deleted successfully.

Example

Delete an existing account by account ID or account alias.

Request

```
curl -X POST http://localhost:9888/delete-account -d '{"account_id": "08F0663C00A02"}'  
# or  
curl -X POST http://localhost:9888/delete-account -d '{"account_alias": "alice"}'
```

Response

No response in case operation was successful.

Create Account Receiver Endpoint

Creates an address and control program.

The address and control program are a one to one relationship.

In the `build-transaction` endpoint, the receiver is the address when the action is of type `control_address`, and the receiver is the control program when the action is of type `control_program`, both can be used to the same effect.

Parameters

Object: `account_alias` | `account_id`

Optional:

- `String` - `account_alias`, alias of account.
- `String` - `account_id`, id of account.

Returns

Object:

- `String` - `address`, address of account.
- `String` - `control_program`, control program of account.

Example

Create an account alias on the existing account ID.

Request

```
curl -X POST http://localhost:9888/create-account-receiver -d '{"account_alias": "alice", "account_id": "0BDQARM800A02"}'
```

Response

```
{
  "address": "ey1q5u8u4el dhjf3lvnkmyl78jj8a75neuryzlknk0",
  "control_program": "0014a70fcae7edbc931fb276d93fe3ca47efa93cf064"
}
```

List Addresses Endpoint

Returns the sub list of all available addresses by account with a limit count.

Parameters

- **String** - **account_alias**, alias of account.
- **String** - **account_id**, id of account.
- **Integer** - **from**, the start position of first address
- **Integer** - **count**, the number of returned

Returns

- **Array of Object**, account address array.
 - **Object**:
 - **String** - **account_alias**, alias of account.
 - **String** - **account_id**, id of account.
 - **String** - **address**, address of account.
 - **Boolean** - **change**, whether the account address is change.

Example

List three addresses from first position by **account_id** or **account_alias**

Request

```
curl -X POST http://localhost:9888/list-addresses -d '{"account_alias": "alice",  
"account_id": "086KQD75G0A02", "from": 0, "count": 3}'
```

Response

```
[  
  {  
    "account_alias": "alice",  
    "account_id": "086KQD75G0A02",  
    "address": "ey1qcn9lf7nxhswratvmg6d78nq7r7yupm36qgs55",  
    "change": false  
  },  
  {  
    "account_alias": "alice",  
    "account_id": "086KQD75G0A02",  
    "address": "ey1qew4h5uvt5ssrtg2alms0j77r94c30m78ucrcxy",  
    "change": false  
  },  
  {  
    "account_alias": "alice",  
    "account_id": "086KQD75G0A02",  
    "address": "ey1qgnp4lte7wge0rsekevjlrdh39vkzz0c2alheue",  
    "change": false  
  }  
]
```

```
}  
]
```

Validate Address Endpoint

Validate that the address is valid and report if it is local or not.

Parameters

Object:

- **string** - **address**, address of account.

Returns

Object:

- **Boolean** - **valid**, whether the account address is valid.
- **Boolean** - **is_local**, whether the account address is local.

Example

Request the validity of an address.

Request

```
curl -X POST http://localhost:9888/validate-address -d '{"address":  
"ey1qcn9lf7nxhswratvmg6d78nq7r7yupm36qgs55"}'
```

Response

```
{  
  "valid": true,  
  "is_local": true,  
}
```

Get Mining Address Endpoint

Query the current mining address.

Parameters

None.

Returns

Object:

- **String** - **mining_address**, the current mining address being used.

Example

Request the current mining address.

Request

```
curl -X POST http://localhost:9888/get-mining-address
```

Response

```
{
  "mining_address": "ey1qnhr65jq3q9gf8uymza8vp0ew8tfyh642wddxh6"
}
```

Set Mining Address Endpoint

Set the current mining address, no matter whether the address is a local one or not. It returns an error message if the address format is incorrect.

Parameters

Object:

- **String** - **mining_address**, mining address to set.

Returns

Object:

- **String** - **mining_address**, the new mining address.

Example

Update the node's mining address.

Request

```
curl -X POST http://localhost:9888/set-mining-address -d
'{"mining_address": "ey1qnhr65jq3q9gf8uymza8vp0ew8tfyh642wddxh6"}'
```


Response

```
{
  "mining_address": "ey1qnhr65jq3q9gf8uymza8vp0ew8tfyh642wddxh6"
}
```

Get Coinbase Arbitrary Endpoint

Get coinbase arbitrary.

Parameters

None.

Returns

Object:

- **String** - **arbitrary**, the arbitrary data append to coinbase, in hexadecimal format. (The full coinbase data for a block will be `0x00&block_height&arbitrary`.)

Example

Query for the coinbase arbitrary.

Request

```
curl -X POST http://localhost:9888/get-coinbase-arbitrary
```

Response

```
{
  "arbitrary": "ff"
}
```

Set Coinbase Arbitrary Endpoint

Set coinbase arbitrary.

Parameters

Object:

- **String** - **arbitrary**, the arbitrary data to be appended to coinbase, in hexadecimal format.

Returns

Object:

- **String** - **arbitrary**, the arbitrary data being appended to coinbase, in hexadecimal format. (The full coinbase data for a block will be `0x008block_height&arbitrary`.)

Example

Set the coinbase arbitrary.

Request

```
curl -X POST http://localhost:9888/set-coinbase-arbitrary -d '{"arbitrary":"ff"}
```

Response

```
{
  "arbitrary":"ff"
}
```

List **pubkeys** Endpoint

Returns the list of all available **pubkeys** by account.

Parameters

Object: **account_alias** | **account_id** | **public_key**

Optional:

- **String** - **account_alias**, alias of account.
- **String** - **account_id**, id of account.
- **string** - **public_key**, public key.

Returns

Object:

- **String** - **root_xpub**, root xpub.
- **Array of Object** - **pubkey_infos**, public key array.
 - **String** - **pubkey**, public key.
 - **Object** - **derivation_path**, derivation path for root xpub.

Example

Query for the list of **pubkeys** by account ID or account alias.

Request

```
curl -X POST http://localhost:9888/list-pubkeys -d '{"account_id": "0G00LLUV00A02"}'
```

Response

```
{
  "pubkey_infos": [
    {
      "derivation_path": [
        "010100000000000000",
        "0100000000000000"
      ],
      "pubkey": "b7730319feac582056379548360da5c08258e248e5c29de08a97a6614df1425d"
    },
    {
      "derivation_path": [
        "010100000000000000",
        "0200000000000000"
      ],
      "pubkey": "5044a0d6113faaf4cb2550f63a820ab579a2af6134e503b76378490d5fe75af4"
    },
    {
      "derivation_path": [
        "010100000000000000",
        "0300000000000000"
      ],
      "pubkey": "ff5c28ce257b25c2a6e172ded490a708a8e654253836d92eb0a68b81ce63bea3"
    }
  ],
  "root_xpub":
  "94a909319eac179f7694b99b8367b9c02b4414b95961e2e3a5bd887e0616af05a7c5e4448df92cd6cdfd8
  2e57cd7aefc1ee0a7fd0d6a2194b5e5faf82556bedc"
}
```

Create Asset Endpoint

Create an asset definition, it prepares for the issuance of an asset.

Parameters

Object:

- **String** - **alias**, name of the asset.

- **Object** - **definition**, definition of asset.

Optional:(please pick one from the following two ways)

- **Array of String** - **root_xpubs**, xpub array.
- **Integer** - **quorum**, the default value is **1**, threshold of keys that must sign a transaction to spend asset units controlled by the account.

or

- **String** - **issuance_program**, user-defined contract program.

Returns

Object:

- **String** - **id**, asset id.
- **String** - **alias**, name of the asset.
- **String** - **issuance_program**, control program of the issuance of asset.
- **Array of Object** - **keys**, information of asset pubkey.
- **String** - **definition**, definition of asset.
- **Integer** - **quorum**, threshold of keys that must sign a transaction to spend asset units controlled by the account.

Example

Create an asset by **xpubs**:

Request

```
curl -X POST http://localhost:9888/create-asset -d '{"alias": "GOLD", "root_xpubs": ["f6a16704f745a168642712060e6c5a69866147e21ec2447ae628f87d756bb68cc9b91405ad0a95f004090e864fde472f62ba97053ea109837bc89d63a64040d5"], "quorum":1}'
```

Response

```
{
  "id": "3c1cf4c9436e3f942cb2f1d70a584f1c61df3697698daccdc89e46f46a003d0",
  "alias": "GOLD",
  "issuance_program": "766baa209683b893483c0a5a317bf9868a8e2a09691f8aa8c1f3e2a7bb62b157e76712e05151ad696c00c0",
  "keys": [
    {
      "root_xpub": "f6a16704f745a168642712060e6c5a69866147e21ec2447ae628f87d756bb68cc9b91405ad0a95f004090"
```

```
e864fde472f62ba97053ea109837bc89d63a64040d5",
  "asset_pubkey":
    "9683b893483c0a5a317bf9868a8e2a09691f8aa8c1f3e2a7bb62b157e76712e012bd443fa7d56a0627df0
    a29dffcdc52641672a0f5cba54d104ad76eb8b8dfc3",
  "asset_derivation_path": [
    "000200000000000000"
  ]
},
"quorum": 1,
"definition": {}
}
```

Create an asset by `issuance_program`:

Request

```
curl -X POST http://localhost:9888/create-asset -d '{"alias":
"TESTASSET","issuance_program":
"20e9108d3ca8049800727f6a3505b3a2710dc579405dde03c250f16d9a7e1e6e78160014c5a5b563c4623
018557fb299259542b8739f6bc20163201e074b22ed7ae8470c7ba5d8a7bc95e83431a753a17465e8673af
68a82500c22741a547a6413000000007b7b51547ac1631a000000547a547aae7cac00c0",
"definition":{"name":"TESTASSET","symbol":"TESTASSET","decimals":8,"description":{}}}'
```

Response

```
{
  "id": "59621aa82c047bd21f73711d4a7905b7a9fbb49bc1a3fdc309b13807cc8b9094",
  "alias": "TESTASSET",
  "issuance_program":
    "20e9108d3ca8049800727f6a3505b3a2710dc579405dde03c250f16d9a7e1e6e78160014c5a5b563c4623
    018557fb299259542b8739f6bc20163201e074b22ed7ae8470c7ba5d8a7bc95e83431a753a17465e8673af
    68a82500c22741a547a6413000000007b7b51547ac1631a000000547a547aae7cac00c0",
  "keys": null,
  "quorum": 0,
  "definition": {
    "decimals": 8,
    "description": {},
    "name": "TESTASSET",
    "symbol": "TESTASSET"
  }
}
```

Get Asset Endpoint

Query asset details by asset ID.

Parameters

Object:

- **String** - **id**, id of asset.

Returns

Object:

- **String** - **id**, asset id.
- **String** - **alias**, name of the asset.
- **String** - **issuance_program**, control program of the issuance of asset.
- **Integer** - **key_index**, index of key for xpub.
- **Integer** - **quorum**, threshold of keys that must sign a transaction to spend asset units controlled by the account.
- **Array of Object** - **xpubs**, pubkey array.
- **String** - **type**, type of asset.
- **Integer** - **vm_version**, version of VM.
- **String** - **raw_definition_byte**, byte of asset definition.
- **Object** - **definition**, description of asset.

Example

Get asset details by asset ID.

Request

```
curl -X POST http://localhost:9888/get-asset -d '{"id":  
"50ec80b6bc48073f6aa8fa045131a71213c33f3681203b15ddc2e4b81f1f4730"}'
```

Response

```
{  
  "alias": "SILVER",  
  "definition": null,  
  "id": "50ec80b6bc48073f6aa8fa045131a71213c33f3681203b15ddc2e4b81f1f4730",  
  "issue_program":  
  "ae2029cd61d9ef31d40af7541f9a50831d6317fdb0870249d0564fcfa9a8f843589c5151ad",  
  "key_index": 1,  
  "quorum": 1,  
  "raw_definition_byte": "",  
  "type": "asset",  
  "vm_version": 1,  
  "xpubs": [  
    ]
```

```
"34b16ee500615cd325f8b84099f83c1ebecaca67977c5dc9b71ae32ceaf18207f996b0a9725b901d3792689b2babcb60febe3b81a684d9b56b65f67f307d453d"
]
```

List Assets Endpoint

Returns the list of all available assets.

Parameters

None.

Returns

- **Array of Object**, asset array.
 - **Object**:
 - **String** - **id**, asset id.
 - **String** - **alias**, name of the asset.
 - **String** - **issuance_program**, control program of the issuance of asset.
 - **Integer** - **key_index**, index of key for xpub.
 - **Integer** - **quorum**, threshold of keys that must sign a transaction to spend asset units controlled by the account.
 - **Array of Object** - **xpubs**, pubkey array.
 - **String** - **type**, type of asset.
 - **Integer** - **vm_version**, version of VM.
 - **String** - **raw_definition_byte**, byte of asset definition.
 - **Object** - **definition**, description of asset.

Example

List all the available assets.

Request

```
curl -X POST http://localhost:9888/list-assets -d '{}'
```

Response

```
[
{
```

```

"alias": "EY",
"definition": {
  "decimals": 8,
  "description": "Eiyaro Official Issue",
  "name": "EY",
  "symbol": "EY"
},
"id": "ffffffffffffffffffffffffffffffffffffffffffffffffffffffffffff",
"issue_program": "",
"key_index": 0,
"quorum": 0,
"raw_definition_byte":
"7b0a202022646563696d616c73223a20382c0a202022646573637269707469666e223a202242797466d2
04f6666696369616c204973737565222c0a2020226e616d65223a202262746d222c0a20202273796d626f6
c223a202262746d220a7d",
"type": "internal",
"vm_version": 1,
"xpubs": null
},
{
  "alias": "SILVER",
  "definition": null,
  "id": "50ec80b6bc48073f6aa8fa045131a71213c33f3681203b15ddc2e4b81f1f4730",
  "issue_program":
"ae2029cd61d9ef31d40af7541f9a50831d6317fdb0870249d0564fcfa9a8f843589c5151ad",
  "key_index": 1,
  "quorum": 1,
  "raw_definition_byte": "",
  "type": "asset",
  "vm_version": 1,
  "xpubs": [
    "34b16ee500615cd325f8b84099f83c1ebecaca67977c5dc9b71ae32ceaf18207f996b0a9725b901d37926
89b2babcb60febe3b81a684d9b56b65f67f307d453d"
  ]
}
]

```

Update Asset Alias Endpoint

Update asset alias by assetID.

Parameters

Object:

- **String** - **id**, id of asset.
- **String** - **alias**, new alias of asset.

Returns

Nothing the asset alias is updated successfully.

Example

Update asset alias.

Request

```
curl -X POST http://localhost:9888/update-asset-alias -d
'{"id":"50ec80b6bc48073f6aa8fa045131a71213c33f3681203b15ddc2e4b81f1f4730",
"alias":"GOLD"}'
```

Response

No response in case operation was successful.

List Balances Endpoint

Returns the list of all available accounts' balances.

Parameters

Optional:

- **String** - **account_id**, account id.
- **String** - **account_alias**, name of account.

Returns

- **Array of Object**, balances owned by the account.
 - **Object**:
 - **String** - **account_id**, account id.
 - **String** - **account_alias**, name of account.
 - **String** - **asset_id**, asset id.
 - **String** - **asset_alias**, name of asset.
 - **Integer** - **amount**, specified asset balance of account.

Example

List all the available accounts' balances.

Request

```
curl -X POST http://localhost:9888/list-balances -d '{}'
```

Response

```
[
  {
    "account_alias": "default",
    "account_id": "0BDQ9AP100A02",
    "amount": 3550800000000,
    "asset_alias": "EY",
    "asset_id": "ffffffffffffffffffffffffffffffffffffffffffffffffffffffffffff"
  },
  {
    "account_alias": "alice",
    "account_id": "0BDQARM800A04",
    "amount": 600000000000,
    "asset_alias": "EY",
    "asset_id": "ffffffffffffffffffffffffffffffffffffffffffffffffffffffffffff"
  }
]
```

List available accounts' balances by a given `account_id`:

Request

```
curl -X POST http://localhost:9888/list-balances -d '{"account_id":"0BDQ9AP100A02"}'
```

Response

```
[
  {
    "account_alias": "default",
    "account_id": "0BDQ9AP100A02",
    "amount": 3550800000000,
    "asset_alias": "EY",
    "asset_id": "ffffffffffffffffffffffffffffffffffffffffffffffffffffffffffff"
  }
]
```

List Unspent Outputs Endpoint

Returns the sub list of all available unspent outputs for all accounts in your wallet.

Parameters

Object:

Optional:

- **String** - **id**, id of unspent output.
- **Boolean** - **unconfirmed**, is include unconfirmed **utxo**
- **Boolean** - **smart_contract**, is contract **utxo**
- **Integer** - **from**, the start position of first **utxo**
- **Integer** - **count**, the number of returned
- **String** - **account_id**, account id.
- **String** - **account_alias**, name of account.

Returns

- **Array of Object**, unspent output array.
 - **Object**:
 - **String** - **account_id**, account id.
 - **String** - **account_alias**, name of account.
 - **String** - **asset_id**, asset id.
 - **String** - **asset_alias**, name of asset.
 - **Integer** - **amount**, specified asset balance of account.
 - **String** - **address**, address of account.
 - **Boolean** - **change**, whether the account address is change.
 - **String** - **id**, unspent output id.
 - **String** - **program**, program of account.
 - **String** - **control_program_index**, index of program.
 - **String** - **source_id**, source unspent output id.
 - **String** - **source_pos**, position of source unspent output id in block.
 - **String** - **valid_height**, valid height.

Example

List all the available unspent outputs:

Request

```
curl -X POST http://localhost:9888/list-unspent-outputs -d '{}'
```

Response

```
[
  {
    "account_alias": "alice",
    "account_id": "0BKBR6VR00A06",
    "address": "ey1qv3htuvug7qdv46ywcvvzytrwrsyg0swltfa0dm",
    "amount": 2000,
    "asset_alias": "GOLD",
    "asset_id": "1883cce6aab82cf9af8cd085a3115dd4a92cdb8e6a9152acd73d7ae4adb9030a",
    "change": false,
    "control_program_index": 2,
    "id": "58f29f0f85f7bd2a91088bcbe536dee41cd0642dfb1480d3a88589bdbfd642d9",
    "program": "0014646ebe3388f01acae88ec318222c6e1c0887c1df",
    "source_id": "5988c1630c1f325e69bb92cb4b19af14286aa107311bc64b8f1a54629a33e0f4",
    "source_pos": 2,
    "valid_height": 0
  },
  {
    "account_alias": "default",
    "account_id": "0BKBR2D2G0A02",
    "address": "ey1qx7ylnhszg24995d5e0nftu9e87kt9vnxcn633r",
    "amount": 624000000000,
    "asset_alias": "EY",
    "asset_id": "ffffffffffffffffffffffffffffffffffffffffffffffffffffffff",
    "change": false,
    "control_program_index": 12,
    "id": "5af9d3c9b69470983377c1fc0c9125c4ac3bfd32c8d505f2a6042aade8503bc9",
    "program": "00143789f9de0242aa52d1b4cbe695f0b93facb2b266",
    "source_id": "233d1dd49e591980f98e11f333c6c28a867e78448e272011f045131df5aa260b",
    "source_pos": 0,
    "valid_height": 12
  }
]
```

List the unspent output matching the given id:

Request

```
curl -X POST http://localhost:9888/list-unspent-outputs -d '{"id":
"58f29f0f85f7bd2a91088bcbe536dee41cd0642dfb1480d3a88589bdbfd642d9"}'
```

Response

```
{
  "account_alias": "alice",
  "account_id": "0BKBR6VR00A06",
  "address": "ey1qv3htuvug7qdv46ywcvvzytrwrsyg0swltfa0dm",
```

```

"amount": 2000,
"asset_alias": "GOLD",
"asset_id": "1883cce6aab82cf9af8cd085a3115dd4a92cdb8e6a9152acd73d7ae4adb9030a",
"change": false,
"control_program_index": 2,
"id": "58f29f0f85f7bd2a91088bcbe536dee41cd0642dfb1480d3a88589bdbfd642d9",
"program": "0014646ebe3388f01acae88ec318222c6e1c0887c1df",
"source_id": "5988c1630c1f325e69bb92cb4b19af14286aa107311bc64b8f1a54629a33e0f4",
"source_pos": 2,
"valid_height": 0
}

```

Backup Wallet Endpoint

Backs up a wallet to an image file, it contains the accounts' image, the assets' image and the keys' image.

Parameters

None.

Returns

Object:

- Object - **account_image**, account image.
- Object - **asset_image**, asset image.
- Object - **key_images**, key image.

Example

Request a backup of the node's wallet information.

Request

```
curl -X http://localhost:9888/backup-wallet -d '{}'
```

Response

```

{
  "account_image": {
    "slices": [
      {
        "account": {
          "type": "account",
          "xpubs": [

```

```

"395d6e0ac25978c3f52f9c7bdfdf75ce6af02639fd7875b4b1f40778ab1120c6dcf461b7ab6fd310983afb54a9a0fb3e09b6ec0d4364c4808c94383d50fb0681"
    ],
    "quorum": 1,
    "key_index": 1,
    "ID": "0CQTA3EOG0A02",
    "Alias": "def"
  },
  "contract_index": 2
}
]
},
"asset_image": {
  "assets": []
},
"key_images": {
  "xkeys": [
    {
      "crypto": {
        "cipher": "aes-128-ctr",
        "ciphertext":
"bf44766fec149478af9500e25ce0a6bc50bb2fa04e40465781da6ff64e9b3a4c9af3d214cd92c5a41d8498db5f4376526740f960ff429b16e52876aec6860e1d",
        "cipherparams": {
          "iv": "1b0fc61ae4dacb15f0f77d2b4ba67635"
        },
        "kdf": "scrypt",
        "kdfparams": {
          "dklen": 32,
          "n": 4096,
          "p": 6,
          "r": 8,
          "salt": "e133b1e7caae771ff1ab34b14824d6e27ef399f2b7ded4ad3500f080ede4a1dd"
        },
        "mac": "bc6bf411fb63e61a17bc15b94f29cf0d5a0f084c328955da1f7e2b26757cfc23"
      },
      "id": "1f40be59-7400-4fdc-b46b-15009f65363a",
      "type": "eiyaro_kd",
      "version": 1,
      "alias": "default",
      "xpub":
"c4ec9bfd5df19d175e17ff7fed89193c37a4a64e1c0928387da01387ca76c3bfd99390e3373ec4d438522cc2d4644214cd2ec3b00965f7a1fa3546809583191c"
    },
    {
      "crypto": {
        "cipher": "aes-128-ctr",
        "ciphertext":
"f0887c8603cbbafc0a66d5b45f71488e089708c7dea4342625a67858a49d6d08c79cd3f1800627e3c8b4668e8df34fcf0be9df5d9d4503acff05373976c312a9",

```

```

    "cipherparams": {
      "iv": "c111b46f9104f49f2c40aedb827e53b5"
    },
    "kdf": "scrypt",
    "kdfparams": {
      "dklen": 32,
      "n": 4096,
      "p": 6,
      "r": 8,
      "salt": "d9ef588b258b111dea1d99a4e4c5a4f968ab69072176bb95b111922e3bbea9e6"
    },
    "mac": "336f5fee643776e139f05ebe5e4f209d992ff97e16b906105fadac9e86133554"
  },
  "id": "611d407c-9e97-4297-a02a-13cd68e47983",
  "type": "eiyaro_kd",
  "version": 1,
  "alias": "def",
  "xpub":
    "395d6e0ac25978c3f52f9c7bdfdf75ce6af02639fd7875b4b1f40778ab1120c6dcf461b7ab6fd310983afb54a9a0fb3e09b6ec0d4364c4808c94383d50fb0681"
}
]
}
}

```

Restore Wallet Endpoint

Restores the wallet by image file.

Parameters

Object:

- **Object** - **account_image**, account image.
- **Object** - **asset_image**, asset image.
- **Object** - **key_images**, key image.

Returns

None if restore of the wallet was successful.

Example

Restore a node's wallet via the image file.

Request

```
curl -X POST http://localhost:9888/restore-wallet -d
```

```
'{"account_image":{"slices":[{"account":{"type":"account","xpubs":["395d6e0ac25978c3f52f9c7bdfdf75ce6af02639fd7875b4b1f40778ab1120c6dcf461b7ab6fd310983afb54a9a0fb3e09b6ec0d4364c4808c94383d50fb0681"],"quorum":1,"key_index":1,"ID":"0CQTA3EOG0A02","Alias":"def"},"contract_index":2}]},{"asset_image":{"assets":[]},"key_images":{"xkeys":{"crypto":{"cipher":"aes-128-ctr","ciphertext":"bf44766fec149478af9500e25ce0a6bc50bb2fa04e40465781da6ff64e9b3a4c9af3d214cd92c5a41d8498db5f4376526740f960ff429b16e52876aec6860e1d","cipherparams":{"iv":"1b0fc61ae4dacb15f0f77d2b4ba67635"},"kdf":"scrypt","kdfparams":{"dklen":32,"n":4096,"p":6,"r":8,"salt":"e133b1e7caae771ff1ab34b14824d6e27ef399f2b7ded4ad3500f080ede4a1dd"},"mac":"bc6bf411fb63e61a17bc15b94f29cf0d5a0f084c328955da1f7e2b26757cfc23"},"id":"1f40be59-7400-4fdc-b46b-15009f65363a","type":"eiyaro_kd","version":1,"alias":"default","xpub":"c4ec9bfd5df19d175e17ff7fed89193c37a4a64e1c0928387da01387ca76c3bfd99390e3373ec4d438522cc2d4644214cd2ec3b00965f7a1fa3546809583191c"},"crypto":{"cipher":"aes-128-ctr","ciphertext":"f0887c8603cbbafc0a66d5b45f71488e089708c7dea4342625a67858a49d6d08c79cd3f1800627e3c8b4668e8df34fcf0be9df5d9d4503acff05373976c312a9","cipherparams":{"iv":"c111b46f9104f49f2c40aedb827e53b5"},"kdf":"scrypt","kdfparams":{"dklen":32,"n":4096,"p":6,"r":8,"salt":"d9ef588b258b111dea1d99a4e4c5a4f968ab69072176bb95b111922e3bbea9e6"},"mac":"336f5fee643776e139f05ebe5e4f209d992ff97e16b906105fadac9e86133554"},"id":"611d407c-9e97-4297-a02a-13cd68e47983","type":"eiyaro_kd","version":1,"alias":"def","xpub":"395d6e0ac25978c3f52f9c7bdfdf75ce6af02639fd7875b4b1f40778ab1120c6dcf461b7ab6fd310983afb54a9a0fb3e09b6ec0d4364c4808c94383d50fb0681"}]}}'
```

Response

No response in case operation was successful.

Rescan Wallet Endpoint

Trigger a rescan of the block information on the wallet.

Parameters

None.

Returns

Nothing if operation was a success.

Example

Request a rescan of the block information on the node.

Request

```
curl -X POST http://localhost:9888/rescan-wallet -d '{}'
```


Response

No response in case operation was successful.

Network Endpoints

These endpoints are available regardless of the wallet being disabled or not.