Web 3.0 Node Engine Service (NES)

FAQs

Issue 01

Date 2024-11-28

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: https://www.huaweicloud.com/intl/en-us/

Contents

1 APIs

1.1 How Do I Determine Whether Flow Control Is Triggered for an API?

If the returned error code is **429**, flow control has been triggered for the API.

1.2 What Are the Flow Control Policies for Full Nodes?

In order to guarantee the stable operation of your full nodes and optimize their performance, Node Engine Service (NES) assigns weights to APIs according to their specifications. When the total weight of all APIs per second surpasses the threshold, flow control is activated.

If your API requests are continuously restricted, it may result in delayed block synchronization and failed transactions. To prevent your services from being affected, you can:

- add nodes
- expand the specifications of existing nodes
- reduce the API calling frequency
- wait for several seconds and try again

Note that for a JSON-RPC batch processing request, the total weight of all methods in the request is calculated. In addition to the preceding ways, you can split methods to call them.

1.3 How Many Methods Can Be Included in a JSON-RPC Batch Request for the Dedicated Version?

Batch requests are a feature of the Ethereum JSON-RPC API, which allows multiple requests to be sent over HTTP or WebSocket. Each request can contain up to 1000 methods.

1.4 How Many WebSocket Connections Can Be Made Each Time?

Dedicated: A maximum of 1000 WebSocket connections can be made at a time.

Shared: A maximum of 2000 WebSocket connections can be made at a time for a DApp.

1.5 How Do I Use HTTP Endpoints and Authentication Credentials to Access Nodes?

You can perform the following operations to access a node using an authentication credential.

Prerequisites

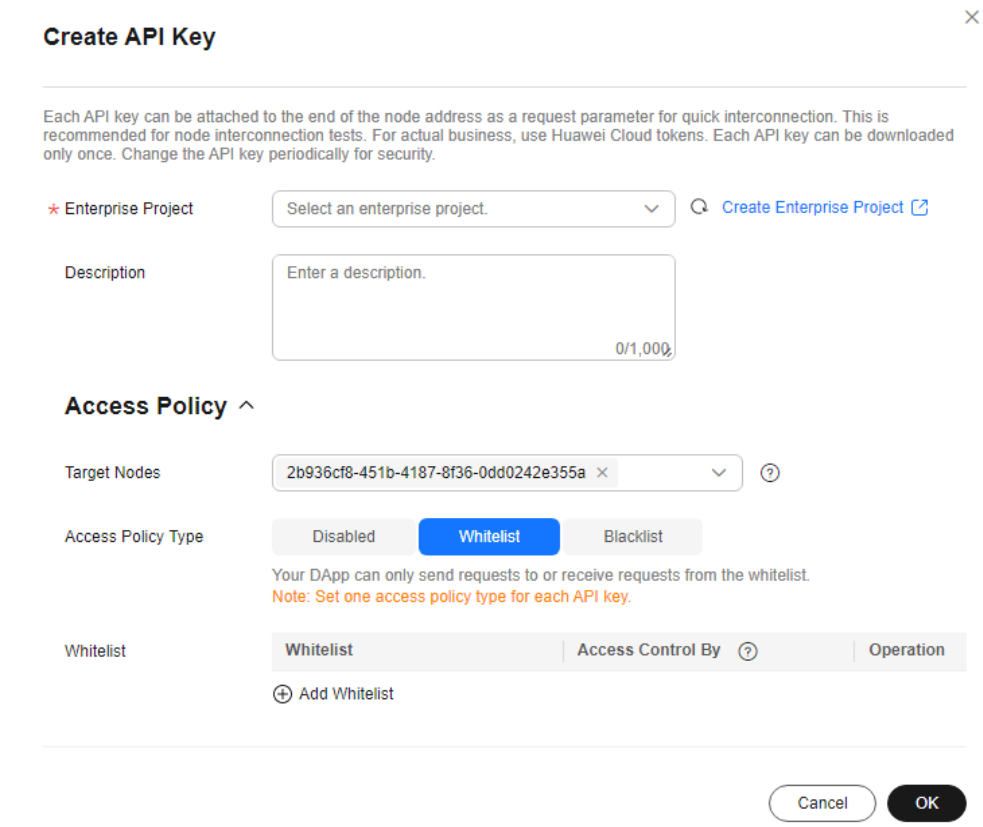
You have created a full node.

Procedure

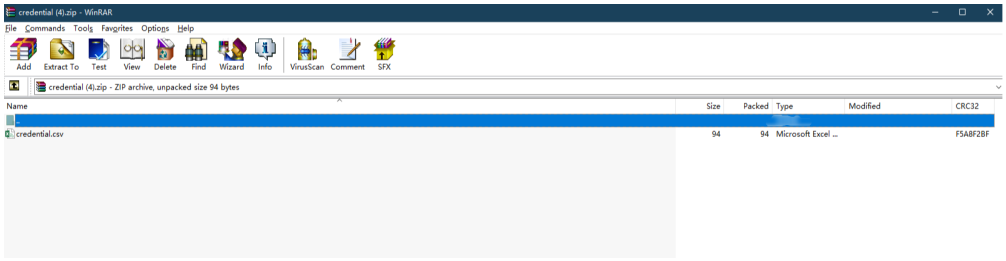
Step 1 Create and obtain an authentication credential.

1. On the NES console, choose **Dedicated** > **Authentication Credential** and click **Create Credential**.
2. Describe the credential and set the access policy.

Figure 1-1 Creating a credential



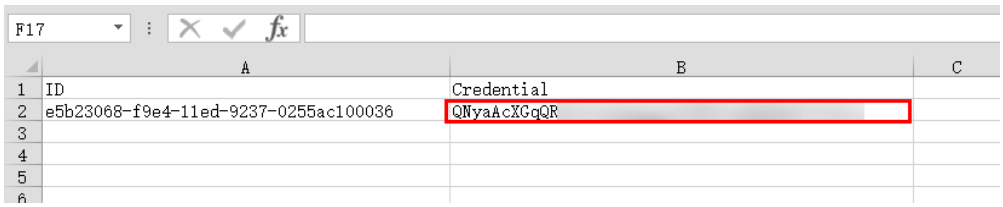
3. Click **OK**. The credential is created and then automatically downloaded as a ZIP package.



NOTE

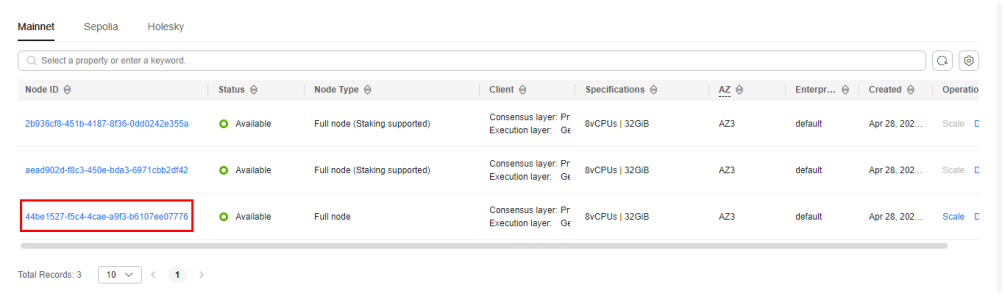
Each credential can be downloaded only once. Change the credential periodically for security.

4. Decompress the package and open the **credential.csv** file to obtain the credential.

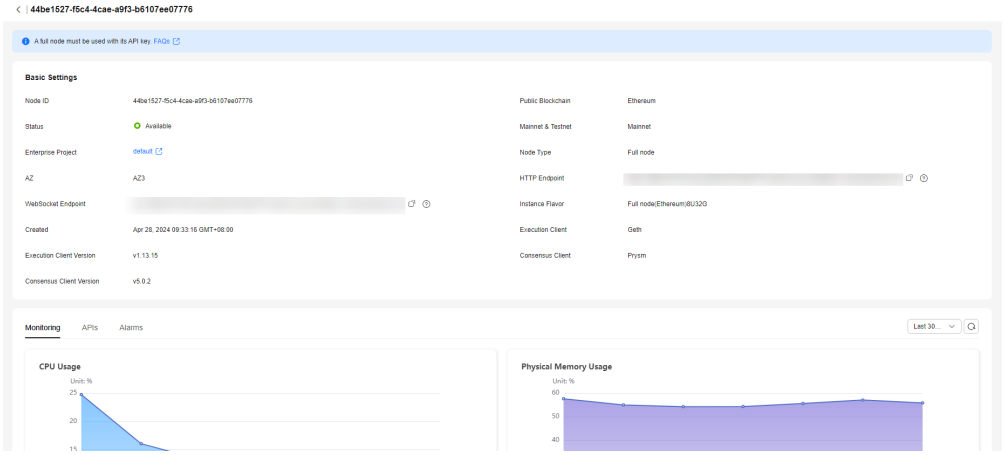


Step 2 Combine a node with the credential.

1. Click a node ID.



2. Obtain the values of **HTTP Endpoint** and **WebSocket Endpoint**.



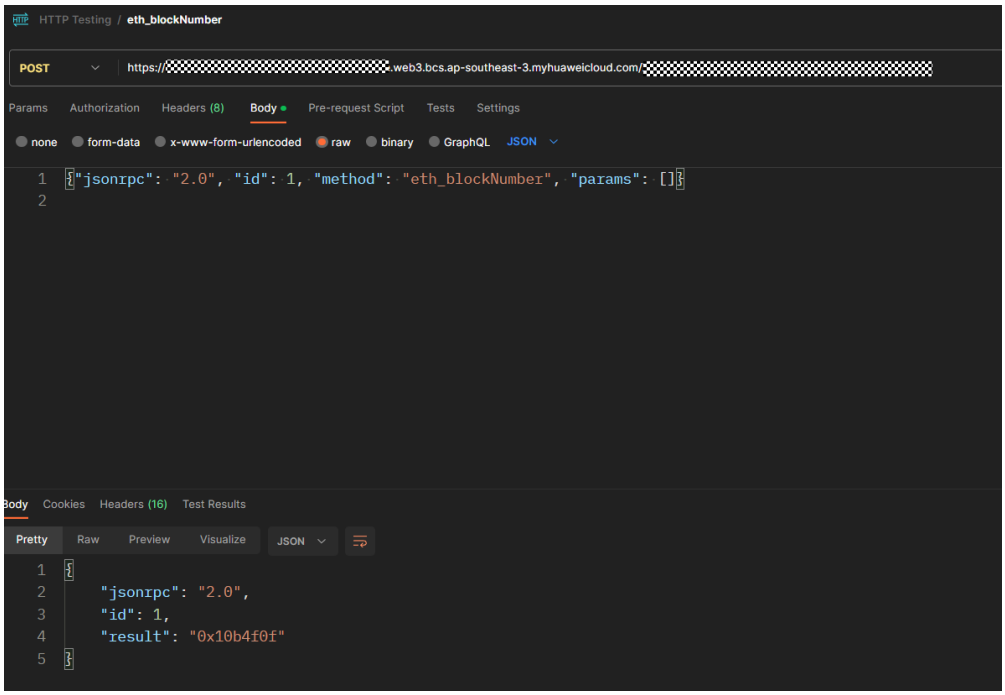
3. Combine the HTTP endpoint or WebSocket endpoint with a credential.

   Specifically:

   - HTTP endpoint: https://*your-http-endpoint*/*your-credential*. For example, https://79b83c56-0a7f-11ee-9cac-0255ac10004e.web3.bcs.ap-southeast-3.myhuaweicloud.com/xxxxxxxxxxxx
   - WebSocket endpoint: wss://*your-http-endpoint*/*your-credential*. For example, wss::://79b83c56-0a7f-11ee-9cac-0255ac10004e.web3.bcs.ap-southeast-3.myhuaweicloud.com/xxxxxxxxxxxx

Step 3 Call the Ethereum node API.

Enter the HTTP endpoint and parameters in Postman and view the returned result.



----End

2 Staking Nodes

2.1 How Many gRPC Connections Can a Staking Node Handle?

500 gRPC connections at most. If there are more connections, the excess connections will time out and the **context deadline exceeded** message will be displayed. In this case, buy more nodes.

2.2 What Are the Default Parameters for Ethereum Nodes?

The following parameters apply to Ethereum nodes:

- **rpc.txfeecap 100**
- **rpc.gascap default**

2.3 How Do I Use a Certificate and an Authentication Credential to Access a Node?

You can perform the following operations to use a certificate and an authentication credential to access a node.

Prerequisites

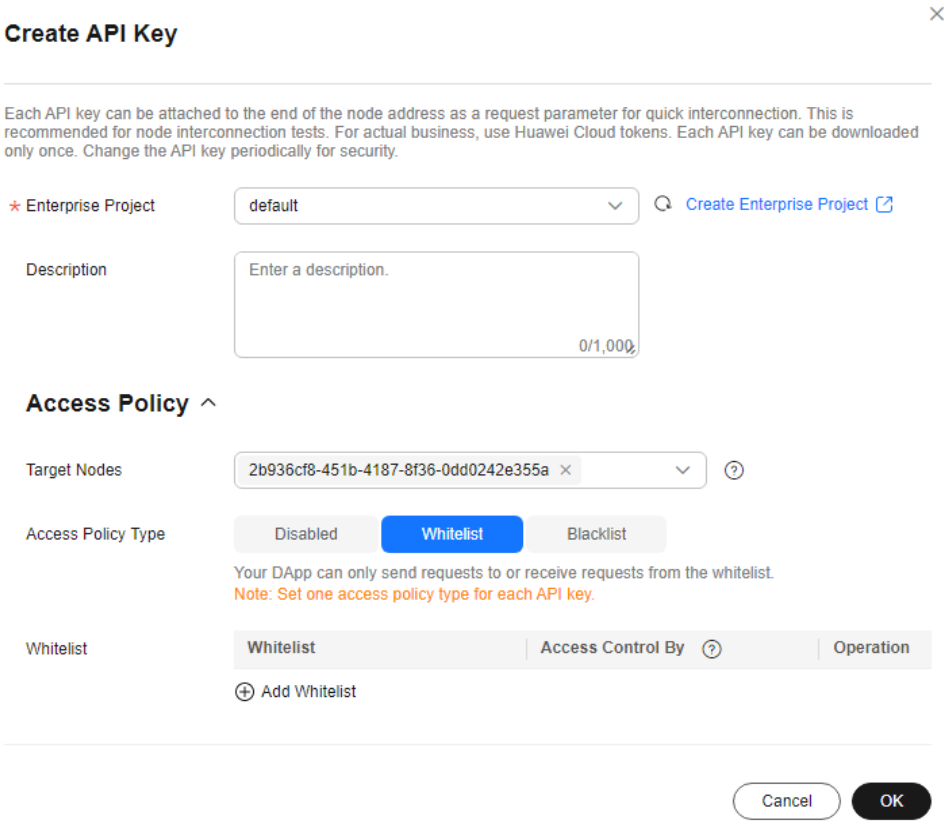
- You have created a full node.
- You have obtained a key on Staking Launchpad. For details, see *NES User Guide (Staking Nodes)*.
- You have downloaded a validator client. Check the **Prysm Documentation** or **Lighthouse Documentation** to download a client as required.

Procedure

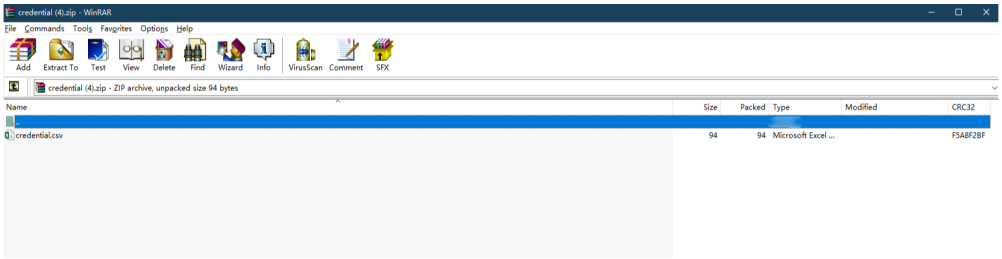
Step 1 Create and obtain an authentication credential.

1. On the NES console, choose **Dedicated** > **Authentication Credential** and click **Create Credential**.
2. Describe the credential and set the access policy.

Figure 2-1 Creating a credential



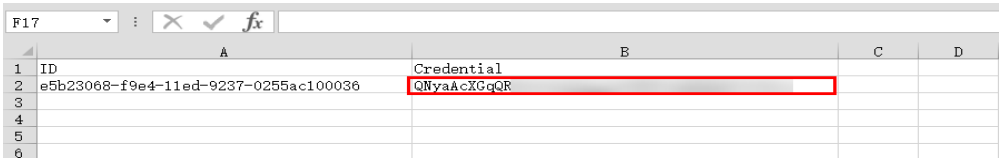
3. Click **OK**. The credential is created and then automatically downloaded as a ZIP package.



NOTE

Each credential can be downloaded only once. Change the credential periodically for security.

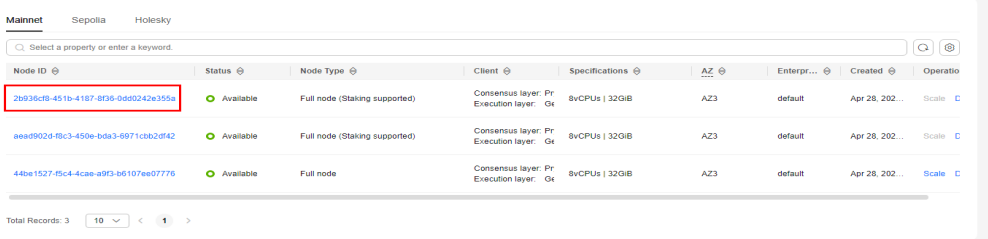
4. Decompress the package and open the **credential.csv** file to obtain the credential.



Step 2 Start a staking node.

1. Click a node ID.

Figure 2-2 Node ID

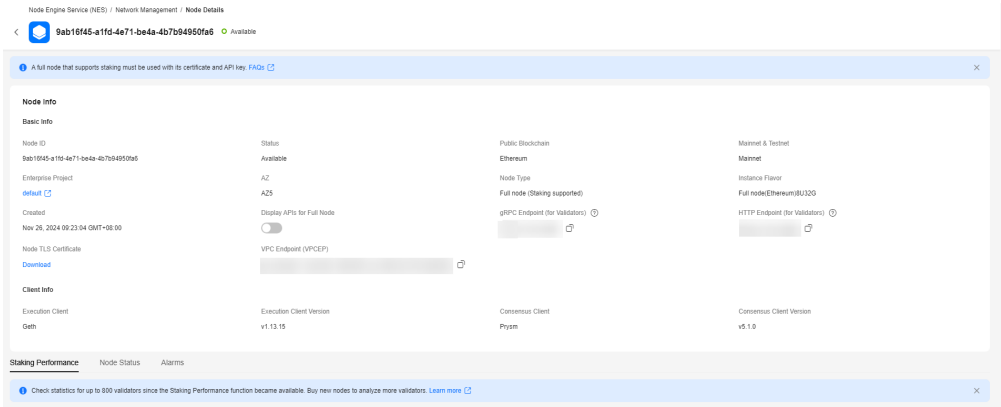


2. Obtain the node information.

   For a Prysm client, you can obtain its **gRPC Endpoint** and **Node TLS Certificate**.

   For a Lighthouse client, you can obtain its **HTTP Endpoint** and **Node TLS Certificate**.

Figure 2-3 Node details of a Prysm client



3. Paste the key and TLS certificate to the hardware machine installed with the script.

   For a Prysm client, run the following command to import the key to the keystore:

   ```
   ./prysm.sh validator accounts import --keys-dir=<YOUR_FOLDER_PATH> --< NETWORK >
   ```

   NETWORK is the staking network and *YOUR_FOLDER_PATH* is the actual key file path.

   For a Lighthouse client, run the following command to import the key to the keystore:

   ```
   lighthouse --network < NETWORK > account validator import --directory < YOUR_FOLDER_PATH >
   ```

   NETWORK is the staking network and *YOUR_FOLDER_PATH* is the actual key file path.

4. After the key is imported, perform the following operations for a Prysm client and Lighthouse client, respectively.

   For a Prysm client, run the **prysm.sh** file, configure the following parameters, and start the staking node.

   - *beacon-rpc-provider*: the value of **gRPC Endpoint**
   - *grpc-headers*: the authentication credential
   - *tls-cert*: the relative path of **Node TLS Certificate**

   Example:

   ```
   ./prysm.sh validator --beacon-rpc-provider=xx.xx.xx.xx:30002 --grpc-headers=credential=xxxxxxxxxxxxxxxxxxxxxx --tls-cert=ca.crt
   ```

   For a Lighthouse client, run the **lighthouse vc** command, configure the following parameters, and start the staking node.

   - *network*: the staking network
   - *suggested-fee-recipient*: the suggested fee recipient
   - *beacon-nodes-tls-certs*: the relative path of **Node TLS Certificate**
   - *beacon-nodes*: the HTTP endpoint or credential information

   ```
   lighthouse vc --network < **NETWORK** > --suggested-fee-recipient <**YourFeeRecipientAddress** > --beacon-nodes-tls-certs ca.pem --beacon-nodes https://xx.xx.xx.xx:30000/xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
   ```

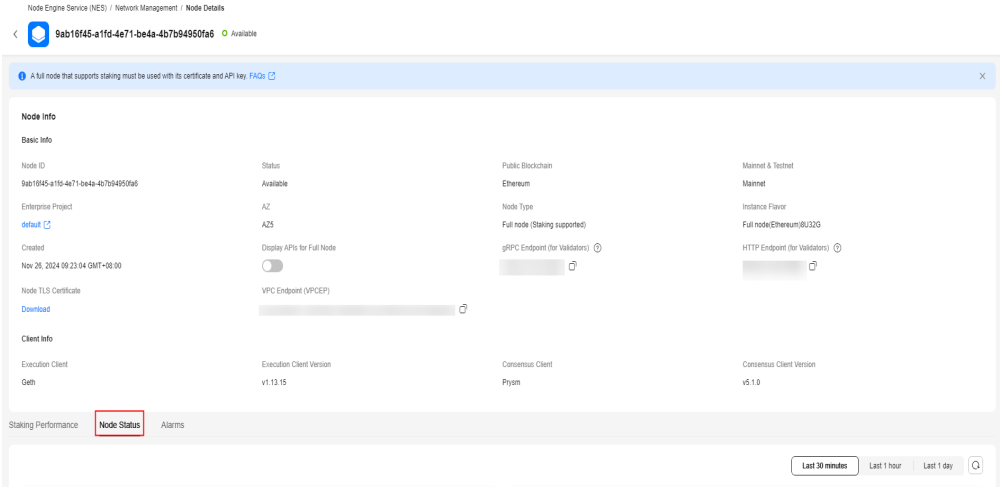
   NOTE

   These parameters are mandatory for interconnecting Huawei Cloud nodes. Check the **Prysm Documentation** and **Lighthouse Documentation** to learn other parameters.

Step 3 Monitor a staking node.

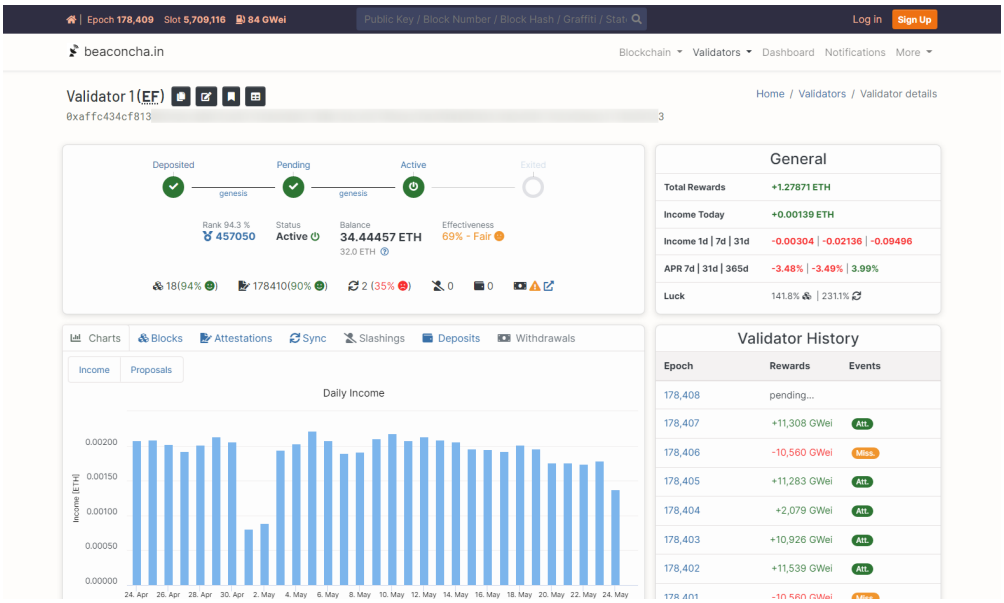
Click a node ID and click the **Node Status** tab page.

Figure 2-4 Node status



NOTE

You need to monitor and perform O&M on the validator client where a staking node has been started. You can also enter the key **on a page similar to the following** to check the client execution.



----End

