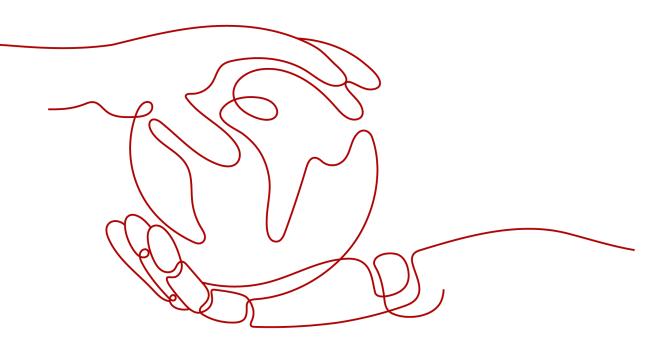
Web 3.0 Node Engine Service (NES)

User Guide (Staking Nodes)

 Issue
 01

 Date
 2024-05-10





Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2024. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Cloud Computing Technologies Co., Ltd.

Trademarks and Permissions

NUAWEI 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 Overview	1
2 Usage Principle	4
3 Staking Operations	6
3.1 Obtaining the Key Using the Staking Launchpad	6
3.2 Creating a Staking Node	
3.3 Creating and Obtaining an API Key	14
3.4 Starting the Staking Node	
3.5 Monitoring Staking Nodes	17
4 VPC Endpoint (VPCEP) Connection	19

Overview

Staking is one of the core mechanisms of Ethereum 2.0. It aims to implement the transition of network consensus algorithm from proof-of-work (PoW) to proof-of-stake (PoS). Staking is the act of depositing 32 ETH to activate validator software. As a validator, you will be responsible for storing data, processing transactions, and adding new blocks to the blockchain. This will keep Ethereum secure for everyone and earn you new ETH in the process.

Benefits of Staking

- Earn rewards: Rewards are given for actions that help the network reach consensus. You will get rewards for running software that properly batches transactions into new blocks and checks the work of other validators because that is what keeps the chain running securely.
- Better security: The Ethereum network gets stronger against attacks as more ETH is staked, as it then requires more ETH to control a majority of the network. Validators are responsible for safeguarding the network and protecting their own interests. This entails that in the event of staking nodes violating regulations or launching network attacks, their ETH will be diminished.
- More sustainable: Stakers do not need to do energy-intensive PoW computations to secure the network, as they rely on staked ETH rather than computing power. This allows Ethereum to efficiently validate and process transactions, resulting in faster overall transaction speeds and throughputs.

Staking Options

Selecting a staking solution depends on how much you are willing to stake. You will need 32 ETH to activate your own validator, but it is possible to stake less.

ltem	Solo Home Staking	Staking as a Service	Pooled Staking
Descr iptio n	Solo staking on Ethereum is the gold standard for staking. It provides full participation rewards, improves the decentralization of the network, and never requires trusting anyone else with your funds. Those considering solo staking should have at least 32 ETH and a dedicated computer connected to the Internet 24/7.	If you do not want or do not feel comfortable dealing with hardware but still want to stake your 32 ETH, this option allows you to delegate the hard part while you earn native block rewards. This option usually walks you through creating a set of validator credentials, uploading your signing keys to them, and depositing your 32 ETH. This allows the service to validate on your behalf. This method of staking requires a certain level of trust in the provider. To limit counter-party risks, the keys to withdrawal your ETH are usually kept in your possession.	Several pooling solutions now exist to assist users who do not have to or do not want to stake 32 ETH. Many of these options include what is known as "liquid staking" which involves an ERC-20 liquidity token that represents your staked ETH. Liquid staking enables easy and anytime exiting and makes staking as simple as a token swap. This option also allows users to hold custody of their assets in their own Ethereum wallet. Note that pooled staking is not native to the Ethereum network.
Rewa rds	Solo stakers receive full rewards directly from the protocol by batching transactions into a new block and checking the work of other validators.	This usually involves full protocol rewards minus monthly fee for node operations. Dashboards are often available to easily track your validator client.	Pooled stakers accrue rewards differently, depending on which method of pooled staking is chosen. Many pooled staking services offer one or more liquidity tokens that represent your staked ETH plus your share of the validator rewards. Liquidity tokens can be held in your own wallet, used in DeFi and sold if you decide to exit.

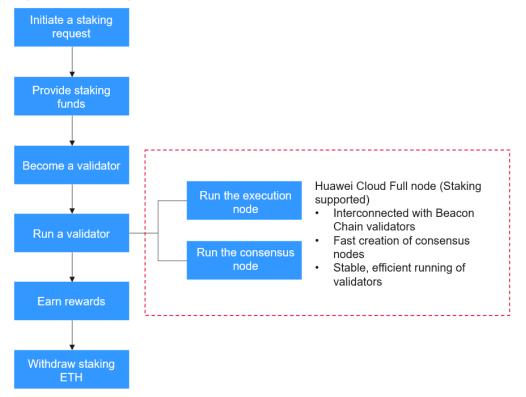
Table 1-1 Staking options

Item	Solo Home Staking	Staking as a Service	Pooled Staking		
Risks	ETH is at stake and cannot be traded. Going offline or other malicious behaviors can result in "slashing" of larger amounts of ETH and forced ejection from the network.	Same risks as solo staking plus counter- party risk of the service provider.	Risks vary depending on the method used. In general, risks consist of a combination of counter-party, smart contract, and execution risk.		
Requi reme nts	 Deposit 32 ETH. Maintain hardware that runs both an Ethereum execution client and consensus client while connected to the Internet. 	 Deposit 32 ETH and generate your keys with assistance. Securely keep your keys. Perform operations according to the service provider. 	 This requires low- ETH commitment as some projects need as little as 0.01 ETH. Deposit directly from your wallet to different pooled staking platforms or simply trade for one of the staking liquidity tokens. 		

2 Usage Principle

Huawei Cloud nodes can be staked separately. To stake a node, you need to purchase a Huawei Cloud staking node, activate a validator, and interconnect the node with the validator. Huawei Cloud keeps the Ethereum nodes running stably by managing the Execution Layer (EL) and Consensus Layer (CL) clients. Note that Huawei Cloud will not keep your keys.

The following figure shows the staking process.





The operations in the red box are performed by NES. Other operations are performed by you.

The following explains the details.

1. Initiate a staking request.

You initiate a staking request and learn about the advisories provided by Ethereum.

2. Provide staking funds.

You deposit a certain amount of ETH to the staking contract. These ETH will support the validator running.

3. Become a validator.

By depositing funds, you will be able to participate in reaching consensus. To become a validator, an individual must stake a specific amount of ETH, for instance, solo staking requires 32 ETH to be staked, and has some technical skills such as knowing how to set up and start a validator client.

4. Run a validator.

You need to verify transaction validity and batch blocks as a validator. To ensure that your validator runs and operates properly, usually, you will need to start an execution node and a consensus node. Huawei Cloud NES provides open gRPC for Beacon Chain validator interconnection. With just a few clicks, EL/CL nodes with the 8 vCPUs | 32 GB flavor can be created effortlessly, eliminating the need for O&M. Additionally, Huawei Cloud-developed algorithms ensure efficient operations of validators.

5. Earn rewards.

Validators will receive block rewards and earn transaction fees. These awards will be allocated based on the staking funds and contribution from the participants.

6. Withdraw staking ETH.

This is optional. The Shanghai/Capella upgrade enabled staking withdrawals on Ethereum.

3 Staking Operations

3.1 Obtaining the Key Using the Staking Launchpad

The Staking Launchpad is an open source application that will help you become a staker. It will guide you through choosing your clients, generating your keys, and depositing your ETH to the staking deposit contract. A checklist is provided to make sure you have covered everything to get your validator set up safely.

Mainnet and Goerli testnet are supported on the Staking Launchpad. It is recommended to test your setup and operational skills on the Goerli testnet.

Step 1 Learn about staking.

The Staking Launchpad provides advisories for you to learn about staking.

_ •	~ 4		•
LIAIIKA		Advac	ORIOC.
Figure	I	AUVIN	
igaic	•	/ (G VIS	or i co

	•	ories	
		OFIOC	
AU			

Everything you should	understand	before beco	ming a validator.

1	Proof of stake	Confirmation
2	Deposit	I have read and agree to the Launchpad terms of service.
3	The terminal	Terms of service
4	Uptime	
5	Bad behavior)
6	Key management)
7	Early adoption risks	I understand and agree to the terms-of-service and everything stated in the previous sections.
8	Checklist	
9	Confirmation	BACK CONTINUE

Step 2 Choose your execution client and consensus client.

You can choose the clients from Eth1 and Eth2 providers and set up your nodes accordingly. If you have purchased Huawei Cloud staking nodes, NES will help you create execution nodes and consensus nodes, so that you can perform staking efficiently.

NOTE

NES uses Geth as the execution client and Prysm and Lighthouse as the consensus client.

Figure 3-2 Choosing an execution client

Choose execution client

Choose your execution client and set up a node

To process incoming validator deposits from the execution layer (formerly 'Eth1' chain), you'll need to run an execution client as well as your consensus client (formerly 'Eth2').



View extensive client comparison >

bug cou	over 33% can cause Ethereum to go offline. If the client has a supermajority (>66%), a Id cause the chain to incorrectly split, potentially leading to slashing.
network	possible, consider running another client at this time to help protect yourself and the .
	est data on network client usage ≯

One of the three original implementations of the Ethereum protocol.

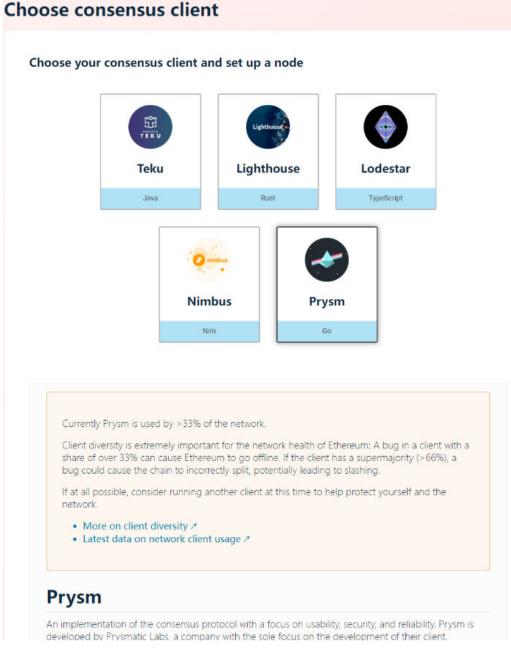


Figure 3-3 Choosing a consensus client

Step 3 Generate keys.

Generate keys using the **key generation tool** provided by Ethereum, and keep the keys safe.

The following shows how to generate keys by downloading the CLI app, that is, the deposit command line interface app.

Figure 3-4 Choosing a key generation tool

Dov	vnload CLI app	Download K	Key Gen GUI app	Build from source	
own	load Wagyu k	(ey Gen ap	р		
e <mark>p 1:</mark> Do	ownload the Wagyu	Key Gen app for	your operating syst	em	
DOWNL	OAD FROM GITHU				
ew Waq					
ew Wag	yu Key Gen audit by H				
ew Wag	yu Key Gen audit by H	lashCloak ≯			
ew Wagy	yu Key Gen audit by H Please make sure t	lashCloak ≯ hat you are dow		ficial StakeHouse GitHu gyu-key-gen/releases	ib account by
	yu Key Gen audit by H Please make sure t	lashCloak ≯ hat you are dow		ficial StakeHouse GitHu gyu-key-gen/releases	ub account by
	yu Key Gen audit by H Please make sure t	lashCloak ≯ hat you are dow			ib account by
	yu Key Gen audit by H Please make sure t	lashCloak ≯ hat you are dow ttps://github.co		gyu-key-gen/releases	

Download the **deposit command line interface** app from GitHub.

Figure 3-5 Downloading the tool

Assets 6		
🗇staking_deposit-cli-d7b5304-darwin-amd64.tar.gz 🛛 🖊 🛛 🗮	9.17 MB	Mar 15
⊗staking_deposit-cli-d7b5304-linux-amd64.tar.gz	20.4 MB	Mar 15
😚 staking_deposit-cli-d7b5304-linux-arm64.tar.gz	19.8 MB	Mar 15
Staking_deposit-cli-d7b5304-windows-amd64.zip	14.2 MB	Mar 15
Source code (zip)		Mar 14
Source code (tar.gz)		Mar 14

Decompress the file you just downloaded. Use the terminal/PowerShell to move into the directory that contains the tool and run the following commands:

Linux/Mac ./deposit new-mnemonic Windows .\deposit.exe new-mnemonic

Then, follow the instructions to generate your keys.

[root@ecs-devnet01 ~]# ./deposit new-mnemonic ***Using the tool on an offline and secure device is highly recommended to keep your mnemonic safe.*** Please choose your language ['1. 2', البرية: ελληνικά', '3. English', '4. Français', '5. Bahasa melayu', '6. Italiano', '7. 日本語', '8. 한국어', '9. Português do Brasil', '10. român', '11. Türkçe', '12. 简体中文']: [English]: Please choose the language of the mnemonic word list ['1. 简体中文', '2. 繁體中文', '3. čeština', '4. English', '5. Italiano', '6. 한국어', '7. Português', '8. Español']: [english]:

Please choose how many new validators you wish to run: 1

Please choose the (mainnet or testnet) network/chain name ['mainnet', 'goerli', 'sepolia', 'zhejiang']: [mainnet]: goerli

Create a password that secures your validator keystore(s). You will need to re-enter this to decrypt them when you set up your Ethereum validators.:

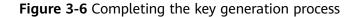
Repeat your keystore password for confirmation:

This is your mnemonic (seed phrase). Write it down and store it safely. It is the ONLY way to retrieve your deposit.

warm merry prevent invalid crash potato tongue portion rapid render chair outdoor summer red chief lesson firm culture infant or island bread fade property

Press any key when you have written down your mnemonic.

Please type your mnemonic (separated by spaces) to confirm you have written it down. Note: you only need to enter the first 4 letters of each word if you'd prefer.





Step 4 Upload deposit data.

Upload the deposit data file you just generated. Ensure that the network appeared in the upper right corner is correct. If you generate a key on the wrong network, the console will prevent you from moving on.

Figure 3-7 Uploading deposit data

💠 Staking Launchpad (Goerli)	Deposit	Clients	Checklist	FAQ	Top Up Withdrawals	Languages Goerli 🗸
					Advisories > Choose client > Generate keys > Upload deposit data > Connect wallet > Summary > Transactions	
					Upload deposit data	
					Upload the deposit data flar you just generated. The (second care beneration) is located in your (bitmy-available value) directory	
					× deposit_data=1684642374.jtcm	
					BACK CONTINUE	

Step 5 Connect to the wallet.

Connect your wallet to the console and make sure you have 32 ETH in your account.

Figure 3-8 Connecting to the wallet

🔶 Staking Launchpad (Goerli)	Deposit	Clients	Checklist	FAQ	Top Up	Withdrawals					Languages Goerli V Ov428_OFAdB
					Advisories >	Choose client >	Generate keys > Upload dep	posit data > Connect wallet	> Summary >	Transactions	
					Conne	ct wallet					
						Metamask		Q,	428620FAdB		
						Ø Network			Goerli testnet		
						Balance		36.8	96 TestnetETH		
							al address for these validators v rol this address before depositi	will be set to 0x4286fadb. Mak ng, as this cannot be changed.	e 100%		
						co	NNECT NEW WALLET	CONTINUE			

Step 6 Confirm the information.

Check the information to ensure it is correct.

Figure 3-9 Confirming the information

Staking Launchpad (Goerli) Deposit Clients Checklist FAQ Top Up Withdrawals	Languages Goeri V outa. raua
Advisories > Choose client > Generat	e keys > Upload deposit data > Connect wallet > Summary > Transactions
Summary	
Launchpad summary Velocitors 1	Total amount required 22TestineETH
Understand the risks	
Indestand that I will not be all Indestand the early adopter Indestand the early adopter Indestand that this transactor	
Make sure you aren't being	phished
Vou are responsible for the transact Fermet Filt to transmission of the transaction to the increased assistance and allowed CHECK DEPOSIT CONT ADDRESS /	
I know how to check that I am set	nding my TestnetETH into the correct deposit contract and will do so.

Step 7 Send the deposit.

Click **CONTINUE** to send the deposit to the Beacon Chain. Now, you have submitted a transaction. Next, you will start the client to complete staking.

Figure 3-10 Sending the deposit

🔶 Staking Launchpad (Goerli)	Deposit	Clients	Checklist	FAQ	Top Up	Withdrawals					Languages Goerli V OH42805Ad8
					Advisories	> Choose client > Generat	e keys > Upload deposit data	> Connect wallet > Summary > *	Transaction	8	
					Trans	actions					
						nfirm deposit nit a transaction to finish your o	leposit.				
						Key list		Action			
						Validator public key 997bd4c9857805ba031e	Status Transaction successful	Action Beaconcha.in / Beaconscan /			
							SEND DEPOSIT				
						e	IACK CONTINU	ε			

NOTE

For details, see Staking Launchpad (Goerli) and Staking Launchpad.

----End

3.2 Creating a Staking Node

- **Step 1** Log in to the NES console.
- Step 2 Click Network Management and click Join Public Blockchain.
- Step 3 Configure parameters.

Figure 3-11 Creating a staking node

	BNB Smart Chain	🔶 Ethereum	Polygon PoS	ờ Tron
* Mainnet & Testnet	Diamet A proof-of-stake (PoS) mainnet for real transactions.	Sepola A proof-of-stake (PoS) testinet with a small amount of data and fast node startup speed.	Holesky The first-ever Ethereum testnet launched on the top of the proof- of-stake (PoS) consensus, serving as a staking, infrastructure, and protocol- developer testnet.	
* Enterprise Project 💿	default	✓ Q Create Enterprise	e Project [/]	
★ Node Type	Full node (Staking supported) Open gRPC for Beacon Chain validat efficiently on Huawei Cloud-develope For details, see Validator APIs [?]	or interconnection. Enjoy fast creation and O dalgorithms.	&M free of EL/CL nodes with the default & vCf	PUs 32 GB flavor. Run your validators
* Consensus Client (?)		Lighthouse Prysm validators of version 4.2.1 and later.		
* Execution Client (?)	Geth			
* Node Specifications	Node Instance Flavor		vCPUs Memory	
	Full node(Ethereum)8U32G	Stable	8vCPUs 32GiB RAM	

Step 4 Click Create.

×

Step 5 Select I have read and agree to the HUAWEI CLOUD User Agreement and Disclaimer. and click Submit.

NOTE

- It takes about 5 to 8 seconds to complete the process.
- Currently, only staking nodes of Ethereum mainnet, Goerli, and Holesky are supported.

----End

3.3 Creating and Obtaining an API Key

- **Step 1** On the NES console, choose **Dedicated** > **API Keys**, then click **Create API Key**.
- Step 2 Describe the API key and set the access policy.

Figure 3-12 Creating an API key

commended for node inte	ned to the end of the node address as a r erconnection tests. For actual business, u key periodically for security.			
Enterprise Project	Select an enterprise project.	~	Q Create Enterp	rise Project 🛽
Description	Enter a description.			
		0/1,000		
Access Policy	^			
Target Nodes	7aa1dfc9-ba49-49ce-a428-ab3a2	d7db0d6 $ imes$	× 0	
Access Policy Type	Disabled Whitelist	Blacklist		
	Your DApp can only send requests Note: Set one access policy type for		from the whitelist.	
Whitelist	Whitelist	Access Contr	ol By 🧿	Operation
	Add Whitelist			

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

는 credential (4).zip - WinRAR						
file <u>C</u> ommands Tools Favorites Options <u>H</u> elp						
Ad Extract To Text View Dette Find Waard Info						
💽 📲 credential (4).zip - ZIP archive, unpacked size 94 bytes						v
Name A	Size	Packed T	ype Modifi	ed	CRC3	2
4) credential.csv	94	94 N	Aicrosoft Excel		F5A8	2BF

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

Step 4 Decompress the package and open the credential.csv file to obtain the API key.

r_1 · · · × · f_x									
В	c	D	E	F	G	H			
Credential									
QNyaAcXGqQR									
	Credential	Credential	Credential	Credential	Credential	Credential			

----End

3.4 Starting the Staking Node

- Step 1 On the NES console, click Network Management.
- **Step 2** Click a node ID.

Figure 3-13 Node ID

Sepolia Holesky Select a property or enter a keyword.								0
Node ID 😔	Status 😔	Node Type \ominus	Client 🖯	Specifications 😔	AZ Θ	Enterpr 😔	Created ⊖	Operatio
7aa1dfc9-ba49-49ce-a428-ab3a2d7db0d6	O Available	Full node	Consensus layer: Pr. Execution layer: Ge	8vCPUs 32GIB	AZ3	default	May 08, 202	Scale D
2b936cf8-451b-4187-8f36-0dd0242e355a	O Available	Full node (Staking supported)	Consensus layer: Pr Execution layer: Ge	8vCPUs 32GiB	AZ3	default	Apr 28, 202	Scale C
aead902d-f8c3-450e-bda3-6971cbb2df42	O Available	Full node (Staking supported)	Consensus layer: Pr Execution layer: Ge	8vCPUs 32GiB	AZ3	default	Apr 28, 202	Scale (
44be1527-f5c4-4cae-a9f3-b5107ee07775	O Available	Full node	Consensus layer: Pr Execution layer: Ge	8vCPUs 32GIB	AZ3	default	Apr 28, 202	Scale D
tal Records: 4 10 V (1)								

Step 3 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 3-14 Node details of a Prysm client

< 2b936ct8-451b-4187-8t	66-0000242e355a							
A full node that supports staking	A full node that supports staking must be used with its centrificate and API key. FriGar 🕑							
Basic Settings								
Node ID	2b936cf8-451b-4187-8f36-0sd0242a355a	Public Blockchain	Ethereum					
Status	Avreilable	Mainnet & Testnet	Mainnet					
Enterprise Project	delauft 🕑	Node Type	Full node (Staking supported)					
AZ	AZ3	Display APIs for Full Node	○ Ciose ∠					
gRPC Endpoint (for Validators)	0° (1)	HTTP Endpoint (for Validators)	0 0					
Node TLS Certificate	Download	Instance Flavor	Full node(Ethereum)8U32G					
Created	Apr 28, 2024 11:25:14 GMT+08:00	Execution Client	Geth					
Execution Client Version	v1.13.15	Consensus Client	Prjsm					
Consensus Client Version	v5.0.2	VPC Endpoint (VPCEP)	ap-toutheast-3 node-2693.fed8dd69-91be-4481-bc4b-9832ae46cddb 🖸					
Monitoring Node Status	Alarms		Last 30					
CPU Usage Unit: % 50		Physical Memory Usage Unit: % 80 70						

Step 4 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.

Step 5 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 API key
- *tls-cert*. the relative path of **Node TLS Certificate**

Example:

./prysm.sh validator --*beacon-rpc-provider*=xx.xx.xx:30002 --*grpc-headers*=credential=xxxxxxxxxxxxxxxxx --*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 API key information

D NOTE

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

----End

3.5 Monitoring Staking Nodes

Step 1 On the NES console, click Network Management.

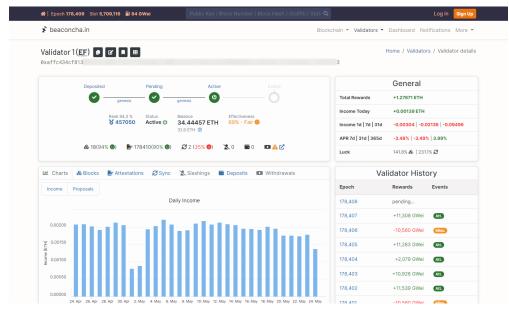
Step 2 Click a node ID and click the **Node Status** tab page.

Figure 3-15 Node status

< 2b936cf8-451b-4187-8	f36-0dd0242e355a								
A full node that supports staki	A flar hode that supports staking must be used with its certificate and API key. (FAD) 🕑								
Basic Settings									
Node ID	2b936cf8-451b-4187-8f36-0dd0242a355a	Public Blockchain	Ethereum						
Status	O Available	Mainnet & Testnet	Mainnet						
Enterprise Project	default 🕑	Node Type	Full node (Staking supported)						
AZ	A23	Display APIs for Full Node	O Close 2						
gRPC Endpoint (for Validators)	0 1	HTTP Endpoint (for Validators)	C (0)						
Node TLS Certificate	Download	Instance Flavor	Full node(Ethereum)8032G						
Created	Apr 28, 2024 11:25:14 GMT+08:00	Execution Client	Geh						
Execution Client Version	v1.13.15	Consensus Client	Prysm						
Consensus Client Version	v5.0.2	VPC Endpoint (VPCEP)	ap southeast-3.node-2b93.fed8dd99-91be-4481-bc4b-9832ae46cddb 🖸						
	-								
Monitoring Node Statu	s Alarms		Last 30 V Q						
Peer Counts	-O- Inbound -O- Outbound	Block Height							
Unit: Times 210		19,829,190							
180 0		10 820 160							

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

4 VPC Endpoint (VPCEP) Connection

VPC endpoints are used for connecting your staking nodes with backend resources, such as Elastic Cloud Server (ECS) and Cloud Container Engine (CCE), through a Huawei Cloud private network.

Prerequisites

You have **created at least one staking node** and enabled **VPC Endpoint (VPCEP)** for it.

Precautions

- The staking node, VPC endpoint, and backend resources must be in the same region, for example, AP-Singapore.
- The VPC endpoint and backend resources must be in the same Virtual Private Cloud (VPC).

Procedure

Step 1 Obtain the VPC endpoint service name.

- 1. Log in to the NES console.
- 2. Choose **Dedicated** > **Network Management**.
- 3. Click a node ID and obtain the **VPC Endpoint** on the node details page.

Basic Settings			
Node ID	2b936cf8-451b-4187-8f36-0dd0242e355a	Public Blockchain	Ethereum
Status	Available	Mainnet & Testnet	Mainnet
Enterprise Project	default (2	Node Type	Full node (Staking supported)
AZ	AZ3	Display APIs for Full Node	○ Close ∠
gRPC Endpoint (for Validators)	190.92.218.36:30036 🖓 💿	HTTP Endpoint (for Validators)	190.92.218.38:30040 🗇 💿
Node TLS Certificate	Download	Instance Flavor	Full node(Ethereum)8U32G
Created	Apr 28, 2024 11:25:14 GMT+08:00	Execution Client	Geth
Execution Client Version	v1.13.15	Consensus Client	Prysm
Consensus Client Version	v5.0.2	VPC Endpoint (VPCEP)	ď

Step 2 Buy a VPC endpoint.

- 1. Log in to the **VPC Endpoint** console.
- 2. Choose VPC Endpoint > VPC Endpoints.

3. Click **Buy VPC Endpoint** and configure parameters. For details, see **Buying a VPC Endpoint**.

< Buy VPC Endpoint ③	
* Region	Artingpore v PArtingpore v Participation v Partic
+ Billing Mode	Figure aux
* Service Category	Cloud service Field a service by name
+ VPC Endpoint Service Name	com my huavisicioud ap-routhess-3 sur
	service name found. Service Type: Interface
	😰 Craelle a Philuide Domain Name 🛈
* VPC	upckcs-d2nd1102.168.0.0 v Q. Wew VPCH
* Subnet	uenetes-epéj(192144) Q Wei Salmeti: Available IP Adverser: 239
★ IPv4 Address	Automatically ansign IP address Manually specify IP address
Access Control	
Tag	h is recommended that you use TMS's predefined tag function to add the same lag to different cloud resources. Vew predefined tags 🔾
	Tag key Tag value
	You can add 20 more tags.
Description	
Price: \$0.014 USD/hour	Not

NOTE

Set **Service Category** to **Find a service by name**. Then, paste the obtained VPC endpoint service name in **VPC Endpoint Service Name**.

- 4. Click **Next**, confirm parameters, and submit the order.
- 5. On the **VPC Endpoints** page, click the ID of the purchased VPC endpoint, and obtain its **Private Domain Name**.

< 15187b72-dc20-477	3-a492-6262c8e8e5cc			0
Summary Access C	ontrol Monitoring Tags			
ID	15187b72-dc20-4773-e492-62d2c8e8e5cc	Status	Accepted	
VPC	vpo-bos-2grl	Type	Interface	
Payer	Service user	VPC Endpoint Service Name	3	
IPv4 Address	192.168.0.58	Created	Apr 15, 2024 20.19:30 GMT+08:00	
Access Control		Private Domain Name		
Description	- 0			•

Step 3 Access the staking node using its port number and the private domain name of the VPC endpoint.

gRPC endpoint: Private network domain name.Port number

HTTP endpoint: Private network domain name.Port number

Obtain the port numbers on the NES console. They are the values of **gRPC Endpoint (for Validators)** and **HTTP Endpoint (for Validators)** on the node details page.

----End