eth_subscribe

Creates a new subscription for particular events. The node returns a subscription ID. For each event that matches the subscription, a notification with relevant data is sent together with the subscription ID.

Parameters

Specify one of the following subscription events:

- newHeads
- : Subscribing to this returns a notification each time a new header is appended to the chain, including chain reorganizations. In a chain reorganization, the subscription emits all new headers for the new chain. Therefore the subscription can emit multiple headers at the same height.
- logs
- : Returns logs that are included in new imported blocks and match the given filter criteria. In case of a chain reorganization, previously sent logs that are on the old chain are resent with the removed property set totrue
- . Logs from transactions that ended up in the new chain are emitted. Therefore a subscription can emit logs for the same transaction multiple times. This parameter has the following fields:
    - address
    - : (optional
    -) Either an address or an array of addresses. Only logs that are created from these addresses are returned.
    - topics
    - : (optional
    -) Only logs that match these specified topics are returned.
- Infura Recommendation
- We strongly recommend specifying a filter (address
- ortopics
- or both) when subscribing to thelogs
- event.

Returns

- subscription ID
- : The ID of the newly created subscription on the node.

Example

ReplaceYOUR-API-KEY with an API key from yourInfura dashboard .

WebSocket connections only Subscription methods are available forWebSocket connections only.

Request

- newHeads
- logs

wscat -c wss://opbnb-mainnet.infura.io/ws/v3/YOUR-API-KEY -x '{"jsonrpc": "2.0", "id": 1, "method": "eth_subscribe", "params": ["newHeads"]}' wscat -c wss://opbnb-mainnet.infura.io/ws/v3/YOUR-API-KEY -x '{"jsonrpc": "2.0", "id": 1, "method": "eth_subscribe", "params": ["logs", {"address": "0x8320fe7702b96808f7bbc0d4a888ed1468216cfd", "topics": ["0xd78a0cb8bb633d06981248b816e7bd33c2a35a6089241d099fa519e361cab902"]}]}'

Response

- New subscription
- newHeads subscription
- logs subscription

{ "id": 1, "jsonrpc": "2.0", "result": "0x9cef478923ff08bf67fde6c64013158d" } { "jsonrpc": "2.0", "method": "eth_subscription", "params": { "result": { "difficulty": "0x15d9223a23aa", "extraData": "0xd98301030584476574887676f312e342e328777696e646f646f646f6f7773", "gasLimit": "0x47e7c4", "gasUsed": "0x38658", "logsBloom": "0x00 "miner": "0xf8b483dba2c3b7176a3da549ad41a48bb3121069", "nonce": "0x084149998194cc5f", "number": "0x1348c9", "parentHash": "0x7736fab79e05dc611604d22470dadad26f56fe494421b5b333de816ce1f25701", "receiptRoot": "0x2fab35823ad00c7bb388595cb46652fe7886e00660a01e867824d3dceb1c8d36", "sha3Uncles": "0x1dcc4de8dec75d7aab85b567b6ccd41ad312451b948a7413f0a142fd40d49347", "stateRoot": "0xb3346685172db67de536d8765c43c31009d0eb3bd9c501c9be3229203f15f378", "timestamp": "0x56ffeff8", "transactionsRoot": "0x0167ffa60e3ebc0b080cdb95f7c0087dd6c0e61413140e39d94d3468d7c9689f" }, "subscription": "0x9ce59a13059e417087c02d3236a0b1cc" } } { "jsonrpc":"2.0", "method":"eth_subscription", "params": { "subscription":"0x4a8a4c0517381924f9838102c5a4dcb7", "result": { "address":"0x8320fe7702b96808f7bbc0d4a888ed1468216cfd","blockHash":"0x61cdb2a09ab99abf791d474f20c2ea89bf8de2923a2d42bb49944c8c993cbf04", "blockNumber":"0x29e87","data":"0x0000000000000000000000000000000000000000000000000000000000000001000000000000000000000000000000000000000000000000000000000000003", "logIndex":"0x0", "topics": ["0xd78a0cb8bb633d06981248b816e7bd33c2a35a6089241d099fa519e361cab902"],"transactionHash":"0xe044554a0a55067caafd07f8020ab9f2af60bdfe337e395ecd84b4877a3d1ab4", "transactionIndex":"0x0" } } }