eth_estimateGas

Generates and returns an estimate of how much gas is necessary to allow the transaction to complete. The transaction will not be added to the blockchain. Note that the estimate may be significantly more than the amount of gas actually used by the transaction, for a variety of reasons including EVM mechanics and node performance.

Gas parameter is capped To prevent abuse of the API, the gas parameter in this eth_estimateGas method and in eth_call is capped at 10x (1000%) the current block gas limit. You can recreate this behavior in your local test environment (besu, geth, or other client) via the rpc.gascap command-line option.

Parameters

- TRANSACTION CALL OBJECT
- [required]
  - from
  - :[optional]
  - 20 Bytes - The address the transaction is sent from.
  - to
  - : 20 Bytes - The address the transaction is directed to.
  - gas
  - :[optional]
  - Hexadecimal value of the gas provided for the transaction execution. eth_estimateGas
  - consumes
  - zero gas, but this parameter may be needed by some executions.
  - gasPrice
  - :[optional]
  - Hexadecimal value of the gas price used for each paid gas.
  - maxPriorityFeePerGas
  - :[optional]
  - Maximum fee, in Wei, the sender is willing to pay per gas above the base fee.
  - See EIP-1559 transactions
  - .
  - maxFeePerGas
  - :[optional]
  - Maximum total fee (base fee + priority fee), in Wei, the sender is willing to pay per gas.
  - See EIP-1559 transactions
  - .
  - value
  - :[optional]
  - Hexadecimal value of the value sent with this transaction.
  - data
  - :[optional]
  - Hash of the method signature and encoded parameters. See the Ethereum contract ABI specification
  - .
  - block number
  - :[required]
  - A string representing a block number, or one of the string tags latest
  - ,earliest
  - ,pending
  - ,safe
  - , or finalized
  - .
  - See the default block parameter
  - .

If no gas limit is specified, geth uses the block gas limit from the pending block as an upper bound. As a result the returned estimate might not be enough to executed the call/transaction when the amount of gas is higher than the pending block gas limit.

Returns

A hexadecimal of the estimate of the gas for the given transaction.

If this call causes the EVM to execute a REVERT operation, an error response of the following form is returned, with the revert reason pre-decoded as a string:

{ "jsonrpc": "2.0", "id": 1, "error": { "code": 3, "message": "execution reverted: Dai/insufficient-balance", "data": "0x08c379a00000000000000000000000000000000000000000000000000000000000000020000000000000000000000000000000000000000000000000000000000000001844461692f696e73756666696
} }

Example

Replace YOUR-API-KEY with an API key from your [Infura dashboard](#) .

Request

- cURL
- WSS

```
curl https://zksync-mainnet.infura.io/v3/YOUR-API-KEY \ -X POST \ -H "Content-Type: application/json" \ -d '{"jsonrpc": "2.0", "method": "eth_estimateGas", "params": [{"from": "0x9cE564c7d09f88E7d8233Cdd3A4d7AC42aBFf3aC", "to": "0xd46e8dd67c5d32be8058bb8eb970870f07244567", "value": "0x9184e72a"}], "id": 1}' wscat -c wss://zksync-mainnet.infura.io/ws/v3/YOUR-API-KEY -x '{"jsonrpc": "2.0", "method": "eth_estimateGas", "params": [{"from": "0x9cE564c7d09f88E7d8233Cdd3A4d7AC42aBFf3aC", "to": "0xd46e8dd67c5d32be8058bb8eb970870f07244567", "value": "0x9184e72a"}], "id": 1}'
```

Response

- JSON

```
{ "jsonrpc" :

"2.0" , "id" :

1 , "result" :

"0x5cec" }
```

Last updated on Nov 5, 2024