

## EthSigner API methods

caution \* All JSON-RPC HTTP examples use the default host and port endpoint `http://127.0.0.1:8545` \*. If using the `--http-listen-host` \* or `--http-listen-port` \* options, update the endpoint. \* The examples use Hyperledger Besu, but any Ethereum client can be used.

## EEA methods

## eea\_sendTransaction

Creates and signs a [private transaction](#) using the [signing key](#).

EthSigner submits the signed transaction to Besu using `eea_sendRawTransaction`.

note Besu uses a Transaction Manager to implement privacy.[Tessera](#) is the Transaction Manager used in this documentation but EthSigner can be used with other Transaction Managers.

## Parameters

Transaction object for private transactions:

Key Type Required/Optional Value from Data, 20bytes Required Address of the sender. Must be the address of the keystore account. to Data, 20bytes Not required for contract creation null for contract creation transaction. Contract address for contract invocation transactions. gas Quantity Optional Gas provided by the sender. Default is90000 . gasPrice Quantity Optional Gas price provided by the sender in Wei. Default is0 . nonce Quantity Optional Number of transactions sent from the account before this one. data Quantity Optional Compiled contract code or hash of the invoked method signature and encoded parameters. privateFrom Data, 20bytes Required Tessera address of the sender privateFor orprivacyGroupId Array of data or data Required Tessera addresses of recipients orprivacy\_group\_id restriction String Required Must be`restricted` tip Submitting a transaction with the same nonce as a pending transaction and a higher gas price replaces the pending transaction with the new one. Use`priv_getTransactionCount` to calculate the nonce.

If not attempting to replace a pending transaction, do not include `nonce` in the private transaction object and nonce management is handled automatically. note If a non-zero value is included in the transaction object, an error is returned. Ether transfers cannot be private transactions.

## Returns

```
result :data - Transaction hash
```

info \* curl with privateFor \* curl with privacy group ID \* JSON result

[illegible]

"2.0" , "id" :

```
1 , "result" :
```

```
"0x6052dd2131667ef3e0a0666f2812db2defceaec91c470bb43de92268e8306778" }
```

## Eth methods

## eth\_accounts

Returns the account address with which EthSigner is signing transactions. That is, the account of the [signing key](#).

Returns multiple accounts if multiple signers are configured.

## Parameters

None

## Returns

Array of data : Account address with which EthSigner is signing transactions.

info \* curl HTTP request \* JSON result

```
curl -X POST --data '{"jsonrpc":"2.0","method":"eth_accounts","params":[],"id":1}' http://127.0.0.1:8545 { "jsonrpc":
```

"2.0" , "id" :

```
1 , "result" :
```

```
[ "0xfe3b557e8fb62b89f4916b721be55ceb828dbd73" ]
```

## eth\_sendTransaction

Creates and signs a transaction using the [signing key](#).

EthSigner submits the signed transaction to Besu using `eth_sendRawTransaction`.

## Parameters

Transaction object:

Key Type Required/Optional Value from Data, 20bytes Required Address of the sender. to Data, 20bytes Optional for contract creation Address of the receiver.null if a contract creation transaction. gas Quantity Optional Gas provided by the sender. Default is90000 . gasPrice Quantity Optional Gas price provided by the sender in Wei. Default is0 . nonce Quantity Optional Number of transactions made by the sender before this one. Must be specified if using[GoQuorum](#) . value Quantity Optional Value transferred in Wei. data Quantity Optional Compiled contract code or hash of the invoked method signature and encoded parameters. tip Submitting a transaction with the same nonce as a pending transaction and a higher gas price replaces the pending transaction with the new one.

## Returns

result :data - 32-byte transaction hash

info \* curl HTTP request \* JSON result

```
curl -X POST --data '{"jsonrpc":"2.0","method":"eth_sendTransaction","params":[{"from": "0xfe3b557e8fb62b89f4916b721be55ceb828dbd73","to": "0xd46e8dd67c5d32be8058bb8eb970870f07244567","gas": "0x7600","gasPrice": "0x9184e72a000","value": "0x9184e72a"}],"id":1}' http://127.0.0.1:8545 { "jsonrpc":
```

"2.0" , "id" :

```
1 , "result" :
```

```
"0x6052dd2131667ef3e0a0666f2812db2defceaec91c470bb43de92268e8306778" }
```

## eth\_sign

Calculates an Ethereum specific signature using `sign(keccak256("\x19Ethereum Signed Message:\n" + len(message) + message)))`.

Adds a prefix to the message that makes the calculated signature recognizable as an Ethereum specific signature. This prevents misuse where a malicious DApp signs arbitrary data (for example a transaction) and uses the signature to impersonate the victim.

### Parameters

DATA : 20-byte account address

DATA : data string to sign

### Returns

DATA : Signature

info \* curl HTTP request \* JSON result

curl -X POST --data '{"jsonrpc":"2.0","method":"eth\_sign","params":["0x78e6e236592597c09d5c137c2af40aec42d12a2", "0x2eadbe1f"], "id":1}' http://127.0.0.1:8545 { "jsonrpc" :

"2.0" , "id" :

1 , "result" :

"0xa6122e277f46fea78f3e97d3354a03ad20b2296733dfefbadc7305c80e70ce9826d44f12ab5aa488689744657491c70d3b654d7f60f8f50beefac9abcf02a4cf1b" }

## eth\_signTransaction

Signs a transaction that can be submitted to Besu at a later time using [eth\\_sendRawTransaction](#) .

### Parameters

Transaction object:

Key Type Required/Optional Value from Data, 20bytes Required Address of the sender. to Data, 20bytes Optional for contract creation Address of the receiver.null if a contract creation transaction. gas Quantity Optional Gas provided by the sender. Default is90000 . gasPrice Quantity Optional Gas price provided by the sender in Wei. Default is0 . nonce Quantity Optional Number of transactions made by the sender before this one. Must be specified if using [GoQuorum](#) . value Quantity Optional Value transferred in Wei. data Quantity Optional Compiled contract code or hash of the invoked method signature and encoded parameters.

### Returns

result :data - The signed transaction object.

info \* curl HTTP request \* JSON result

curl -X POST --data '{"jsonrpc":"2.0","method":"eth\_signTransaction","params":[{"from": "0xfe3b557e8fb62b89f4916b721be55ceb828dbd73","to":

"0xd46e8dd67c5d32be8058bb8eb970870f07244567","gas": "0x7600","gasPrice": "0x9184e72a000","value": "0x9184e72a", "nonce": "0x46"}], "id":1}' http://127.0.0.1:8545 { "jsonrpc" :

"2.0" , "id" :

1 , "result" :

"0xa3f20717a250c2b0b729b7e5becbfff67fdae7e0699da4de7ca5895b02a170a12d887fd3b17bfdce3481f10bea41f45ba9f709d39ce8325427b57afcf994cee1b" } [Edit this page](#) Last updatedonMar 30, 2023 byEric Lin [Previous Subcommands](#) [Next Multikey parameters](#)