D4D4D4;--ch-t-background: #1E1E1E;--ch-t-lighter-inlineBackground: #1e1e1ee6;--ch-t-editor-background: #1E1E1E;--ch-t-editor-foreground: #D4D4D4;--ch-t-editor-rangeHighlightBackground: #ffffff0b;--ch-t-editor-infoForeground: #3794FF;--ch-t-editor-selectionBackground: #264F78;--ch-t-focusBorder: #007FD4;--ch-t-tab-activeBackground: #1E1E1E;--ch-t-tab-activeForeground: #ffffff;--ch-t-tab-inactiveBackground: #2D2D2D;--ch-t-tab-inactiveForeground: #ffffff80;--ch-t-tab-border: #252526;--ch-t-tab-activeBorder: #1E1E1E;--ch-t-editorGroup-border: #444444;--ch-t-editorGroupHeader-tabsBackground: #252526;--ch-t-editorLineNumber-foreground: #858585;--ch-t-input-background: #3C3C3C;--ch-t-input-foreground: #D4D4D4;--ch-t-icon-foreground: #C5C5C5;--ch-t-sideBar-background: #252526;--ch-t-sideBar-foreground: #D4D4D4;--ch-t-sideBar-border: #252526;--ch-t-list-activeSelectionBackground: #094771;--ch-t-list-activeSelectionForeground: #fffffe;--ch-t-list-hoverBackground: #2A2D2E; }

Send Transactions

In this guide, you will learn how to create Safe transactions, sign them, collect the signatures from the different owners, and execute them.

For more detailed information, see theStarter Kit Reference .

Prerequisites

- Node.js and npm(opens in a new tab)

Install dependencies

First, you need to install some dependencies.

```
_10 pnpm add @safe-global/starter-kit
```

Steps

Imports

Here are all the necessary imports for this guide.

```
_10 import { createSafeClient } from '@safe-global/sdk-starter-kit'
```

Create a signer

Firstly, you need to get a signer, which will be the owner of a Safe account after it's deployed.

This example uses a private key, but any way to get an EIP-1193 compatible signer can be used.

```
_10 const SIGNER_ADDRESS = // ... _10 const SIGNER_PRIVATE_KEY = // ... _10 const RPC_URL =
'https://rpc.ankr.com/eth_sepolia'
```

Initialize theSafeClient

New Safe account Existing Safe account When deploying a new Safe account, you need to pass the configuration of the Safe in thesafeOptions property. The Safe account is configured with your signer as the only owner in this case.

```
_10 const safeClient = await createSafeClient({ _10 provider: RPC_URL, _10 signer: SIGNER_PRIVATE_KEY, _10
safeOptions: { _10 owners: [SIGNER_ADDRESS], _10 threshold: 1 _10 } _10 })
```

Create a Safe transaction

Create an array of Safe transactions to execute.

```
_10 const transactions = [{ _10 to: '0x...', _10 data: '0x', _10 value: '0' _10 }]
```

Send the Safe transaction

If you configured your Safe withthreshold equal to1 , calling thesend method will execute the Safe transaction. However, if thethreshold is greater than1 the other owners of the Safe will need to confirm the transaction until the required number of signatures are collected.

```
_10 const txResult = await safeClient.send({ transactions }) _10 _10 const safeTxHash = txResult.transactions?.safeTxHash
```

Confirm the Safe transaction

If the Safe transaction needs to be confirmed by other Safe owners, call theconfirm method from a newSafeClient instance initialized with each of the signers that need to confirm it.

```
_10 const newSafeClient = await createSafeClient({ _10 provider: RPC_URL, _10 signer, _10 safeAddress: '0x...' _10 })
```

Finally, retrieve all the pending Safe transactions from the Safe Transaction Service and confirm the one you just created with each missing owner.

```
_10 const pendingTransactions = await newSafeClient.getPendingTransactions() _10 _10 for (const transaction of
pendingTransactions.results) { _10 if (transaction.safeTxHash !== safeTxHash) { _10 return _10 } _10 _10 const txResult =
await safeClient.confirm({ safeTxHash }) _10 }
```

Recap and further reading

After following this guide, you are able to deploy new Safe accounts and create, sign, and execute Safe transactions with the Starter Kit.

Starter Kit Send User Operations Was this page helpful?

Report issue