

Sending a Transaction

[Suggest Edits](#)

To send a transaction on the EVM, you must have the provider, the connected wallet, and a valid transaction object. We can make use of the [ethers.js](#) frontend library to help with building out our transaction.

Below is a code example that builds out a valid transaction object and prompts the transaction to send with permission from the signer.

```
JavaScript import { TransactionResponse, Web3Provider } from "@ethersproject/providers"; import { utils } from "ethers";
```

```
/* * Send 1 wei from the connected wallet to itself * @param {Web3Provider} provider a web3 provider * @returns {Promise} a raw transaction object / export async function sendTransaction( provider: Web3Provider ): Promise { try { const signer = provider.getSigner(); const address = await signer.getAddress(); const gasPrice = await provider.getGasPrice(); const transactionParameters = { nonce: await provider.getTransactionCount(address), gasPrice, gasLimit: utils.hexlify(100000), to: address, from: address, value: utils.parseUnits("1", "wei"), }; return signer.sendTransaction(transactionParameters); } catch (error) { console.warn(error); throw new Error("Error sending transaction"); } } And that's it! Call this function with the provider you created earlier and you've officially sent a transaction on the EVM with the ME wallet. Please reference our demo if you have any questions on how to put together a full project that natively connects to the ME wallet. Updated about 1 month ago
```