Инструкция по Bridge для фронта

Делаем unlock кошелька Metamask или какого-то другого.

При переводе из ETH в BSC:

1. Загружаем контракт ZAM токена на Эфире, используя его ABI (zam\_eth.json) и адрес (0xd373576a9e738F37Dc6882328358Ff69c4cAf4c6)
2. Проверяем allowance, для этого используем метод контракта токена allowance(address owner, address spender)

Пример кода:

allowance\_sum = await token\_contract.methods.allowance(address owner, address spender).call()

В нашем случае:

owner - адрес пользователя кошелька

spender - адрес контракта агента ETHAgent в Эфире (0x0d02e94fB14f19b5E04E4CC91f62c1A80B09a6B6)

1. Если allowance\_sum > 0, то переходим к пункту 4. Если равняется 0, то кнопка должна называться “Approve”, при нажатии кнопки вызываем метод контракта токена approve(address spender, uint256 amount)

Пример кода:

const transactionParameters = {

to: contractAddress,

from: window.ethereum.selectedAddress,

'data': token\_contract.methods.approve(address spender, uint256 amount).encodeABI()

};

В нашем случае:

to - адрес контракта zam-токена в Эфире (0xd373576a9e738F37Dc6882328358Ff69c4cAf4c6)

from - адрес пользователя кошелька

Поле ‘data’:

spender - адрес контракта агента в Эфире(0x0d02e94fB14f19b5E04E4CC91f62c1A80B09a6B6)

amount - количество денег, которые пользователь разрешает потратить (можно указать сразу большое значение 10\*\*60)

Этот метод вызываем, как в обучении в этой ссылке <https://docs.alchemy.com/alchemy/tutorials/nft-minter#upload-the-metadata-to-ipfs>

1. После того, как получили allowance или он уже был, можем выполнить swap. Кнопка должна называться “Swap” или “Transfer”. Загружаем контракт агента ETHAgent в Эфире, используя его abi (eth\_agent.json) и адрес (0x0d02e94fB14f19b5E04E4CC91f62c1A80B09a6B6)
2. После того, как пользователь заполнит поле Amount и нажмет кнопку, необходимо вызвать метод контракта агента swapETH2BSC(address token, uint256 amount)

Пример кода:

const transactionParameters = {

to: contractAddress,

from: window.ethereum.selectedAddress,

'data': agent\_contract.methods.swapETH2BSC(address token, uint256 amount).encodeABI()

};

В нашем случае:

to - адрес контракта агента (0x0d02e94fB14f19b5E04E4CC91f62c1A80B09a6B6)

from - адрес пользователя кошелька

Поле ‘data’:

token - адрес zam-токена в Эфире (0xd373576a9e738F37Dc6882328358Ff69c4cAf4c6)

amount - количество токенов, которые пользователь хочет перевести

При переводе из BSC в Эфир:

1. Загружаем контракт ZAM токена на BSC, используя его ABI (zam\_bsc.json) и адрес (0xBbcF57177D8752B21d080bf30a06CE20aD6333F8)
2. Проверяем allowance, для этого используем метод контракта токена allowance(address owner, address spender)

Пример кода:

allowance\_sum = await token\_contract.methods.allowance(address owner, address spender).call()

В нашем случае:

owner - адрес пользователя кошелька

spender - адрес контракта агента BSCAgent в Binance (0xa89f6B45D48A925B75697882CC28Eed955b59E84)

1. Если allowance\_sum > 0, то переходим к пункту 4. Если равняется 0, то кнопка должна называться “Approve”, при нажатии кнопки вызываем метод контракта токена approve(address spender, uint256 amount)

Пример кода:

const transactionParameters = {

to: contractAddress,

from: window.ethereum.selectedAddress,

'data': token\_contract.methods.approve(address spender, uint256 amount).encodeABI()

};

В нашем случае:

to - адрес контракта zam-токена в Binance (0xBbcF57177D8752B21d080bf30a06CE20aD6333F8)

from - адрес пользователя кошелька

Поле ‘data’:

spender - адрес контракта агента в Binance (0xa89f6B45D48A925B75697882CC28Eed955b59E84)

amount - количество денег, которые пользователь разрешает потратить (можно указать сразу большое значение 10\*\*60)

Этот метод вызываем, как в обучении в этой ссылке <https://docs.alchemy.com/alchemy/tutorials/nft-minter#upload-the-metadata-to-ipfs>

1. После того, как получили allowance или он уже был, можем выполнить swap. Кнопка должна называться уже “Swap” или “Transfer”. Загружаем контракт агента BSCAgent в Binance, используя его abi (bsc\_agent.json) и адрес (0xa89f6B45D48A925B75697882CC28Eed955b59E84)
2. После того, как пользователь заполнит поле Amount и нажмет кнопку, необходимо вызвать метод контракта агента swapBSC2ETH(address token, uint256 amount)

Пример кода:

const transactionParameters = {

to: contractAddress,

from: window.ethereum.selectedAddress,

'data': agent\_contract.methods.swapBSC2ETH(address token, uint256 amount).encodeABI()

};

В нашем случае:

to - адрес контракта агента (0xa89f6B45D48A925B75697882CC28Eed955b59E84)

from - адрес пользователя кошелька

Поле ‘data’:

token - адрес zam-токена в Binance (0xBbcF57177D8752B21d080bf30a06CE20aD6333F8)

amount - количество токенов, которые пользователь хочет перевести

Примечания:

1. Нужны abi смарт-контрактов: zam\_eth.json, eth\_agent.json, zam\_bsc,json, bsc\_agent.json
2. Адреса смартов:

Zam-токен в Эфире: 0xd373576a9e738F37Dc6882328358Ff69c4cAf4c6

Zam-токен в Binance: 0xBbcF57177D8752B21d080bf30a06CE20aD6333F8

Агент ETHAgent в Эфире: 0x0d02e94fB14f19b5E04E4CC91f62c1A80B09a6B6

Агент BSCAgent в Binance: 0xa89f6B45D48A925B75697882CC28Eed955b59E84

1. Для загрузки смарт-контракта можно использовать метод:

window.contract = await new web3.eth.Contract(contractABI, contractAddress)

contractABI - json-файл контракта

contractAddress - адрес контракта в сети

Полезные ссылки:

1. <https://docs.alchemy.com/alchemy/tutorials/nft-minter#upload-the-metadata-to-ipfs> - Кусок кода для отправки транзакции
2. <https://web3js.readthedocs.io/en/v1.2.11/web3-eth-contract.html#methods-mymethod-call> - Документация по методу call()