

Koi Paymaster Service

Integrate the Mute Paymaster into your dApp By utilizing [zkSync Era account abstraction paymasters](#) , Mute allows for gas to be paid with other tokens instead of ETH. Instead of paying a transaction fee of 0.40 with ETH, the user can opt in to use any of the supported tokens below. The conversion price is based on the token/ETH LP pool on the Mute Switch. This is a free to use service.

Integration

You can easily integrate the Mute Paymaster into your dApp - [Official documentation](#)

[Official Paymaster Address](#) const

PAYMASTER_ADDRESS

=

'0x4ae2Ba9A5C653038C6d2F5D9F80B28011A454597' ... // approve the token for spending via the paymaster ... const
greeting =

"a new greeting" ; const tx =

await greeter . populateTransaction . setGreeting (greeting); const gasPrice =

await sender . provider . getGasPrice (); const gasLimit =

await greeter . estimateGas . setGreeting (greeting); const paymasterParams = utils . getPaymasterParams (testnetPaymaster ,

{ type :

"ApprovalBased" , TOKEN_ADDRESS , minimalAllowance : ethers . BigNumber . from (1),, innerInput :

new

Uint8Array (), }); const sentTx =

await sender . sendTransaction ({ ... tx , maxFeePerGas : gasPrice , maxPriorityFeePerGas : BigNumber . from (0), gasLimit , customData :

{ gasPerPubdata : utils . DEFAULT_GAS_PER_PUBDATA_LIMIT , paymasterParams , }, });

Supported Tokens

* * [USDC](#) * * * [WBTC](#) * * * [MUTE](#) * [Protocol - Previous VWAP Oracles Next - Koi DAO veKOI](#) Last modified 2mo ago

On this page Integration Supported Tokens