Default EVM Adapter

[@web3auth/default-evm-adapter](#)

[â](#)

Default EVM adapter allows you to create external adapters like [@web3auth/torus-evm](#) , [@web3auth/metamask](#) , and [@web3auth/wallet-connect-v2](#) easily with a single line of code.

Basic Details[â](#)

Adapter Name:default-evm-adapter

[â](#)

Package Name:[@web3auth/default-evm-adapter](#)

[â](#)

authMode:DAPP

[â](#)

chainNamespace:EIP155

[â](#)

Default:NO

[â](#)

Installation[â](#)

- npm
- Yarn
- pnpm

npm install --save @web3auth/default-evm-adapter yarn add @web3auth/default-evm-adapter pnpm add @web3auth/default-evm-adapter

Arguments[â](#)

- Table
- Interface

Parameter type options Web3AuthNoModalOptions export

declare

const

getDefaultExternalAdapters :

(params :

{ options :

Web3AuthNoModalOptions ; })

=>

Promise < IAdapter < unknown

[ ]

;

Example[a]

```
import

{ getDefaultExternalAdapters }

from

"@web3auth/default-evm-adapter" ; import

{

EthereumPrivateKeyProvider

}

from

"@web3auth/ethereum-provider" ;

const chainConfig =

{ chainId :

"0x1" ,

// Please use 0x1 for Mainnet rpcTarget :

"https://rpc.ankr.com/eth" , displayName :

"Ethereum Mainnet" , blockExplorerUrl :

"https://etherscan.io/" , ticker :

"ETH" , tickerName :

"Ethereum" , logo :

"https://images.toruswallet.io/eth.svg" , } ;

const privateKeyProvider =

new

EthereumPrivateKeyProvider ( { config :

{ chainConfig }

} ) ;

const web3AuthOptions :

Web3AuthOptions

=

{ clientId , chainConfig :

{

... chainConfig , chainNamespace :

CHAIN_NAMESPACES . EIP155

} , web3AuthNetwork :

WEB3AUTH_NETWORK . SAPPHIRE_MAINNET , privateKeyProvider : privateKeyProvider , } ;

const adapters =

await

getDefaultExternalAdapters ( { options : web3AuthOptions } ) ;
```

```
adapters . forEach ( ( adapter )

=>

{ web3auth . configureAdapter ( adapter ) ; } )
```