

Common Provider for PnP Web SDKs

[@web3auth/base-provider](#)

â

For connecting to blockchains other than EVM, Solana, and XRPL, you need to use the private key from Web3Auth and manually make RPC calls to the blockchain. This flow is facilitated by @web3auth/base-provider package.

In this section, we'll explore more about how you can use this provider with our SDKs.

Installationâ

[@web3auth/base-provider](#)

â

```
npm install --save @web3auth/base-provider
```

Initialisationâ

Import the `CommonPrivateKeyProvider` class from @web3auth/base-provider .

```
import
```

```
{
```

```
  CommonPrivateKeyProvider
```

```
}
```

```
from
```

```
"@web3auth/base-provider" ;
```

Assign the `CommonPrivateKeyProvider`

class to a variableâ

After creating your Web3Auth instance, you need to initialize the `CommonPrivateKeyProvider` and add it to a class for further usage.

```
const privateKeyProvider =
```

```
new
```

```
CommonPrivateKeyProvider ( ) ;
```

Note * The common private key provider only exposes one RPC method (i.e. 'private_key') to get the * private key of the logged-in user.

Chain Configurationâ

```
const chainConfig =
```

```
{ chainNamespace :
```

```
"other" , chainId :
```

```
"Algorand" ,
```

```
// rpcTarget :
```

```
"https://api.algoexplorer.io" , // Avoid using public rpcTarget in production. // Use services like PureStake, AlgoExplorer API, etc. displayName :
```

```
"Algorand Mainnet" , blockExplorerUrl :
```

```
"https://algoexplorer.io" , ticker :
```

"ALGO" , tickerName :

"Algorand" , } ; * PnP Modal SDK * PnP NoModal SDK * CoreKit SFA Web SDK

import

{

Web3Auth

}

from

"@web3auth/modal" ; import

{

CommonPrivateKeyProvider

}

from

"@web3auth/base-provider" ; import

{

WEB3AUTH_NETWORK ,

CHAIN_NAMESPACES

}

from

"@web3auth/base" ;

const privateKeyProvider =

new

CommonPrivateKeyProvider ({ config :

{ chainConfig : chainConfig } , }) ;

const web3auth =

new

Web3Auth ({ // Get it from Web3Auth Dashboard clientId , web3AuthNetwork :

WEB3AUTH_NETWORK . SAPPHIRE_MAINNET , privateKeyProvider : privateKeyProvider , }) ; import

{

Web3AuthNoModal

}

from

"@web3auth/no-modal" ; import

{

AuthAdapter

}

from

"@web3auth/auth-adapter" ; import

```

{
  CommonPrivateKeyProvider
}

from
"@web3auth/base-provider" ; import
{
  WEB3AUTH_NETWORK ,
  CHAIN_NAMESPACES
}

from
"@web3auth/base" ;
const privateKeyProvider =
new
CommonPrivateKeyProvider ( { config :
{ chainConfig } , } ) ;
const web3auth =
new
Web3AuthNoModal ( { clientId ,
// Get it from Web3Auth Dashboard web3AuthNetwork :
WEB3AUTH_NETWORK . SAPPHIRE_MAINNET , } ) ;
const authAdapter =
new
AuthAdapter ( ) ; web3auth . configureAdapter ( authAdapter ) ; import
{
  Web3Auth
}

from
"@web3auth/single-factor-auth" ; import
{
  CommonPrivateKeyProvider
}

from
"@web3auth/base-provider" ; import
{
  WEB3AUTH_NETWORK ,
  CHAIN_NAMESPACES
}

```

```

from
"@web3auth/base" ;

const web3auth =
new
Web3Auth ( { clientId ,
// Get your Client ID from Web3Auth Dashboard web3AuthNetwork :
WEB3AUTH_NETWORK . SAPPHIRE_MAINNET , } ) ;

const privateKeyProvider =
new
CommonPrivateKeyProvider ( { config :
{ chainConfig } , } ) ;

```

Usage

On connection, you can use this provider as a private key provider to expose the user's private key in the frontend context

```

//Assuming the user is already logged in. async

getPrivateKey ( )

{ const privateKey =

await web3authSfaprovider . request ( { method :

"private_key" } ) ; //Do something with privateKey } tip Please refer to our Non EVM Connect Blockchain Reference for more
information.

```

Examples

[### Integrate PnP No Modal SDK with Algorand SAMPLE APP Use Algorand with Plug and Play No Modal SDK Source Code Guide](#) pnp web @web3auth/no-modal javascript algorand
[### Integrate PnP No Modal SDK with Aptos SAMPLE APP Use Aptos with Plug and Play No Modal SDK Source Code Guide](#) pnp web @web3auth/no-modal javascript aptos ed25519
[### Integrate PnP No Modal SDK with Cosmos SAMPLE APP Use Cosmos with Plug and Play No Modal SDK Source Code Guide](#) pnp web @web3auth/no-modal javascript cosmos
[### Integrate PnP No Modal SDK with ImmutableX SAMPLE APP Use ImmutableX with Plug and Play No Modal SDK Source Code Guide](#) pnp web @web3auth/no-modal javascript immutablex
[### Integrate PnP No Modal SDK with Multiple Chains SAMPLE APP Use Multiple Chains at once with Plug and Play No Modal SDK Source Code Guide](#) pnp web @web3auth/no-modal javascript multi chain evm solana tezos starknet polkadot
[### Integrate PnP No Modal SDK with Polkadot SAMPLE APP Use Polkadot with Plug and Play No Modal SDK Source Code Guide](#) pnp web @web3auth/no-modal javascript polkadot
[### Integrate PnP No Modal SDK with Polymesh SAMPLE APP Use Polymesh with Plug and Play No Modal SDK Source Code Guide](#) pnp web @web3auth/no-modal javascript polymesh
[### Integrate PnP No Modal SDK with StarkEx SAMPLE APP Use StarkEx with Plug and Play No Modal SDK Source Code Guide](#) pnp web @web3auth/no-modal javascript starkex
[### Integrate PnP No Modal SDK with StarkNet SAMPLE APP Use StarkNet with Plug and Play No Modal SDK Source Code Guide](#) pnp web @web3auth/no-modal javascript starknet
[### Integrate PnP No Modal SDK with Sui SAMPLE APP Use Sui with Plug and Play No Modal SDK Source Code Guide](#) pnp web @web3auth/no-modal javascript sui
[### Integrate PnP No Modal SDK with Tezos SAMPLE APP Use Tezos with Plug and Play No Modal SDK Source Code Guide](#) pnp web @web3auth/no-modal javascript tezos
[Edit this page](#) [Previous XRPL Provider](#) [Next Wagmi Connector](#)