

Multi Factor Authentication in PnP Flutter SDK

At Web3Auth, we prioritize your security by offering Multi-Factor Authentication (MFA). MFA is an extra layer of protection that verifies your identity when accessing your account. To ensure ownership, you must provide two or more different backup factors. You have the option to choose from the device, social, backup factor (seed phrase), and password factors to guarantee access to your Web3 account. Once you create a recovery factor, MFA is enabled, and your keys are divided into three shares for off-chain multi-sig, making the key self-custodial. With backup factors, you can easily recover your account if you lose access to your original device or helps login into a new device.

Minimum Growth plan required This is a paid feature and the minimum [pricing plan](#) to use this SDK in a production environment is the Growth Plan . You can use this feature in the development environment for free.

Using themfaLevel

to enable MFA [^](#)

For a dApp, we provide various options to set up Multi-Factor Authentication. You can customize the MFA screen by setting themfaLevel argument. You can enable or disable a backup factor and change their order. Currently, there are four values formfaLevel :

- default
 - : presents the MFA screen every third login
- optional
 - : presents the MFA screen on every login, but you can skip it
- mandatory
 - : make it mandatory to set up MFA after login
- none
 - : skips the MFA setup screen

Note If you are using default verifiers, your users may have set up MFA on other dApps that also use default Web3Auth verifiers. In this case, the MFA screen will continue to appear if the user has enabled MFA on other dApps. This is because MFA cannot be turned off once it is enabled. Web3AuthFlutter . login (LoginParams (loginProvider :

Provider . google , mfaLevel :

MfaLevel . MANDATORY ,) ,) ; Usage Future < void

initWeb3Auth ()

async

{ HashMap themeMap =

HashMap < String ,

String

() ; themeMap ['primary']

=

"#229954" ;

Uri redirectUrl ; if

(Platform . isAndroid)

{ redirectUrl =

Uri . parse ('{SCHEME}://{HOST}/auth') ; // w3a://com.example.w3aflutter/auth }

else

if

(Platform . isIOS)

{ redirectUrl =

Uri . parse ('{bundleId}://auth') ; // com.example.w3aflutter://openlogin }

```

else
{ throw
UnknownException ( 'Unknown platform' ) ; }

await

Web3AuthFlutter . init ( Web3AuthOptions ( clientId :
"WEB3AUTH_CLIENT_ID" , network :
Network . sapphire_mainnet , redirectUrl : redirectUrl , ) , ) ; }

final

Web3AuthResponse response =

await

Web3AuthFlutter . login ( LoginParams ( loginProvider :
Provider . google , mfaLevel :
MFALevel . MANDATORY , ) , ) ;

```

Using themfaSettings

to configure MFA Order[â](#)

You can configure the order of MFA, and enable or disable MFA type by passing themfaSettings object inWeb3AuthOptions .

Minimum SCALE plan required This is a paid feature and the minimum[pricing plan](#) to use this SDK in a production environment is theSCALE Plan . You can use this feature in the development environment for free. MfaSettings

- Table
- Class

Parameter Description deviceShareFactor? MFA setting for deviceShareFactor. It acceptsMfaSetting as a value. backUpShareFactor? MFA setting for backUpShareFactor. It acceptsMfaSetting as a value. socialBackupFactor? MFA setting for socialBackupFactor. It acceptsMfaSetting as a value. passwordFactor? MFA setting for passwordFactor. It acceptsMfaSetting as a value. class

MfaSettings

```

{ final

MfaSetting ? deviceShareFactor ; final

MfaSetting ? backUpShareFactor ; final

MfaSetting ? socialBackupFactor ; final

MfaSetting ? passwordFactor ;

MfaSettings ( { this . deviceShareFactor , this . backUpShareFactor , this . socialBackupFactor , this . passwordFactor , } ) ;

Map < String ,

dynamic

toJson ( )

{ return

{ 'deviceShareFactor' : deviceShareFactor , 'backUpShareFactor' : backUpShareFactor , 'socialBackupFactor' :
socialBackupFactor , 'passwordFactor' : passwordFactor , } ; } } MfaSetting

```

- Table
- Class

Parameter Description enable Enable/Disable MFA. It acceptsbool as a value. priority? Priority of MFA. It acceptsint as a

value, where valid range is from 1 to 4. mandatory? Mandatory/Optional MFA. It accepts bool as a value. class

MfaSetting

```
{ final bool enable ; final int ? priority ; final bool ? mandatory ;
```

```
MfaSetting ( { required this . enable ,
```

```
this . priority ,
```

```
this . mandatory } ) ;
```

```
Map < String ,
```

```
dynamic
```

```
toJson ( )
```

```
{ return
```

```
{ 'enable' : enable ,
```

```
'priority' : priority ,
```

```
'mandatory' : mandatory } ; } } Future < void
```

```
initWeb3Auth ( )
```

```
async
```

```
{
```

```
late final
```

```
Uri redirectUrl ; if
```

```
( Platform . isAndroid )
```

```
{ redirectUrl =
```

```
Uri . parse ( '{SCHEME}://{HOST}/auth' ) ; // w3a://com.example.w3aflutter/auth }
```

```
else
```

```
{ redirectUrl =
```

```
Uri . parse ( '{bundleId}://auth' ) ; // com.example.w3aflutter://auth }
```

```
await
```

```
Web3AuthFlutter . init ( Web3AuthOptions ( clientId :
```

```
"WEB3AUTH_CLIENT_ID" , network :
```

```
Network . sapphire_mainnet , redirectUrl : redirectUrl , mfaSettings :
```

```
MfaSettings ( deviceShareFactor :
```

```
MfaSetting ( enable :
```

```
true , priority :
```

```
4 , mandatory :
```

```
false , ) , backUpShareFactor :
```

```
MfaSetting ( enable :
```

```
true , priority :
```

```
2 , mandatory :
```

```
true , ) , socialBackupFactor :
```

```
MfaSetting ( enable :  
true , priority :  
3 , mandatory :  
false , ) , passwordFactor :
```

```
MfaSetting ( enable :  
true , priority :  
1 , mandatory :  
true , ) , ) , ) ;
```

```
await
```

```
Web3AuthFlutter . initialize ( ) ; }Edit this page Previous Custom Authentication Next dApp Share
```