Using Core Kit SFA iOS SDK

After successfully installing and initializing SingleFactorAuth, you can use it to authenticate your users and obtain their private and public keys.

The SingleFactorAuth instance natively provides the following functions:

- getKey()
-
    - Returns the torus key using theverifier
- ,verifierId
- &idToken
- .

getKey()

â

getKey()

To obtain a user's Torus key using the Web3Auth SFA iOS SDK, you can call thegetKey() function.

Variable Description loginParams Parameters for the login. It takesLoginParams as the value.

Returnsâ

public

func

getKey (loginParams :

LoginParams)

async

throws

->

TorusKey

LoginParams

â

getKey(loginParams: LoginParams)

- Table
- Type

Parameter Description verifier The verifier obtained from the Web3Auth Dashboard. It's a mandatory field and takesString as a value. verifierId The verifierId used for the verification. It takesString as a value. idToken The idToken of the user obtained from login provider. It takesString as a value. subVerifierInfoArray? Sub verifier info. Usually used during aggregate verifier. It takesArray as a value. public

class

LoginParams

{ public

let verifier :

String public

let verifierId :

String public

let idToken :

String public

let subVerifierInfoArray :

[TorusSubVerifierInfo] ?

public

init (verifier :

String , verifierId :

String , idToken :

String)

{ self . verifier = verifier self . verifierId = verifierId self . idToken = idToken self . subVerifierInfoArray =

nil }

init (verifier :

String , verifierId :

String , idToken :

String , subVerifierInfoArray :

[TorusSubVerifierInfo])

{ self . verifier = verifier self . verifierId = verifierId self . idToken = idToken self . subVerifierInfoArray = subVerifierInfoArray } } Usage let loginParams =

LoginParams (verifier :

"YOUR_VERIFIER_NAME" , verifierId :

"YOUR_VERIFIER_ID" , idToken :

"YOUR_ID_TOKEN") let torusKey =

try

await singleFactoreAuth . getKey (loginParams : loginParams) NOTE Web3Auth SFA iOS SDK only works for users who havenot enabled MFA . MFA enabled users For MFA enabled users, you'll see an Error message.

Exampleâ

import

SingleFactorAuth import

JWTDecode

/ . . ./

let jwt =

try

decode (jwt : id_token) let result =

try

await

SingleFactorAuth (singleFactorAuthArgs :

. init (network :

. CYAN)) . getKey (loginParams :

. init (verifier :

"web3auth-firebase-examples" , verifierId : jwt . subject ??

"YOUR_VERIFIER_ID" , idToken : id_token))

await

MainActor . run (body :

{ privateKey = result . getPrivateKey ( ) publicAddress = result . getPublicAddress ( ) })

Session Managementâ

We have also included Session Management in this SDK, so calling the initialize function to get the TorusKey value without re-logging in the user if a user has an active session will return the TorusKey struct; otherwise, it will return nil.

if

let savedKey =

try

await singleFactoreAuth . initialize ( )

{ print (savedKey . getPrivateKey ( )) print (savedKey . getPublicAddress ( )) }

Edit this page Previous Authentication Next Overview