

useBatchSession

[@biconomy/use-aa](#) • Hook

[@biconomy/use-aa](#) / useBatchSession

Description

Uses a previously created batch session ([see here](#)) which batches transactions in the context of a users smart account.

Parameters

type

Transaction

=

{ to :

string ; value : BigNumberish |

string ; data :

string ; } ;

type

UseBatchSessionProps

=

{ / The BuildUserOpOptions options. See <https://bcnmy.github.io/biconomy-client-sdk/types/BuildUserOpOptions.html> for further detail */ options ? : BuildUserOpOptions ; / The transactions to be batched. / transactions : Transaction | Transaction [] ; /An array of indexes for the transactions corresponding to the relevant session IDs. / correspondingIndexes :

number [] ; } ;

Returns

[userOpReceipt](#)

type

UserOpResponse

=

{ userOpHash :

string ; wait (_confirmations ? :

number) :

Promise < UserOpReceipt

; waitForTxHash () :

Promise < UserOpStatus

; } ;

Example

import

{ useBatchSession , useUserOpWait ,

```

Options
}

from
"@biconomy/useAA" ; import
{ polygonAmoy }
from
"viem/chains" ; import
{ encodeFunctionData , parseAbi }
from
"wagmi" ;

const
UseBatchSession
=
( { smartAccountAddress } )
=>
{ const
{ mutate , data : userOpResponse , error , isPending }
=
useBatchSession ( ) ;

const
{ isLoading : waitIsLoading , isSuccess : waitIsSuccess , error : waitError , data : waitData , }
=
useUserOpWait ( userOpResponse ) ;

const nftMintTx :
Transaction
=
{ to :
"0x1758f42Af7026fBbB559Dc60EcE0De3ef81f665e" , data :
encodeFunctionData ( { abi :
parseAbi ( [ "function safeMint(address _to)" ] ) , functionName :
"safeMint" , args :
[ smartAccountAddress ] , } ) , } ;

const
txTwice
=
( )
=> mutate ( { transactions :

```

```

[ nftMintTx , nftMintTx ] , correspondingIndexes :
[ 0 ,
1 ] , options :
Options . Sponsored , } ) ;
useEffect ( ( )
=>
{ if
( waitIsSuccess && waitData ?. success ===
"true" )
{ console . log ( "Successful mint: "
+ { polygonAmoy . blockExplorers . default . url } /tx/ { waitData ?. receipt ?. transactionHash } ) ; } } ,
[ waitIsSuccess ] ) ;
return
( < ErrorGuard

```

errors

```

{ [ error , waitError ] }

< Button title = " Use Session to Mint Twice " onClickFunc = { txTwice } isLoading = { isPending || waitIsLoading }
/> </ ErrorGuard

) ; } ;

```

Source

[hooks/useBatchSession.ts:91](#) [Previous BiconomyProvider](#) [Next useCreateBatchSession](#)