

React Native Storage Module - tKey JS SDK

[@tkey/react-native-storage](https://github.com/tkey/react-native-storage)

[â](#)

The React Native Storage Module helps you store and recall key shares from the React Native Encrypted Store/ Expo Secure Store.

Installation[â](#)

- npm
- Yarn
- pnpm

npm install --save @tkey/react-native-storage yarn add @tkey/react-native-storage pnpm add @tkey/react-native-storage

Initialization[â](#)

- Bare React Native
- Expo React Native

```
import
{
  ReactNativeStorageModule
}
from
"@tkey/react-native-storage" ; import
EncryptedStorage
from
"react-native-encrypted-storage" ; const reactNativeStorageModule =
new
ReactNativeStorageModule ( EncryptedStorage ) ; import
{
  ReactNativeStorageModule
}
from
"@tkey/react-native-storage" ; import
*
as
SecureStore
from
"expo-secure-store" ; const reactNativeStorageModule =
new
ReactNativeStorageModule ( SecureStore ) ;
```

Usage[â](#)

With the `ReactNativeStorageModules`, you have access to the multiple functions as mentioned in the type reference, however, the most important ones are:

Get ShareStore from Storage

`getStoreFromReactNativeStorage(): Promise;`

Example

```
const share =  
await ( tKeyInstance . modules . reactNativeStorage  
as  
ReactNativeStorageModule ) . getStoreFromReactNativeStorage ( ) ;
```

Store Device Share

`storeDeviceShare(deviceShareStore: ShareStore, customDeviceInfo?: StringifiedType): Promise;`

This function takes the following arguments:

- `deviceShareStore`
- `: TheShareStore`
- object to store.
- `customDeviceInfo?`
- `: Optional`, Information about the device to store.

Example

```
const generateShareResult =  
await tKeyInstance . generateNewShare ( ) ; const share =  
await tKeyInstance . outputShareStore ( generateShareResult . newShareIndex ) ; await ( tKeyInstance . modules .  
reactNativeStorage  
as  
ReactNativeStorageModule ) . storeDeviceShare ( share ) ;
```

Type Reference

ReactNativeStorageModule

[â](#)

export

default

class

ReactNativeStorageModule

implements

IModule

{ moduleName :

string ; tbSDK :

ITKeyApi ; private keyStore ; constructor (storage :

SecureStore

|

```
EncryptedStorage ) ; setModuleReferences ( tbSDK :
```

```
ITKeyApi ) :
```

```
void ; initialize ( ) :
```

```
Promise < void
```

```
    ; storeDeviceShare ( deviceShareStore :
```

```
ShareStore , customDeviceInfo ? :
```

```
StringifiedType ) :
```

```
Promise < void
```

```
    ; storeShareOnReactNativeStorage ( share :
```

```
ShareStore ) :
```

```
Promise < void
```

```
    ; getStoreFromReactNativeStorage ( ) :
```

```
Promise < ShareStore
```

```
    ; inputShareFromReactNativeStorage ( ) :
```

```
Promise < void
```

```
    ; }
```

ShareStore

[â](#)

```
class
```

```
ShareStore
```

```
implements
```

```
ISerializable
```

```
{ share :
```

```
Share ; polynomialID :
```

```
PolynomialID ; constructor ( share :
```

```
Share , polynomialID :
```

```
PolynomialID ) ; static
```

```
fromJSON ( value :
```

```
StringifiedType ) :
```

```
ShareStore ; toJSON ( ) :
```

```
StringifiedType ; } interface
```

```
ISerializable
```

```
{ toJSON ( ) :
```

```
StringifiedType ; } Edit this page Previous Web Storage Next Chrome Storage
```