
Account Abstraction

Release 0.1

Dror Tirosh

Nov 29, 2022

CONTENTS

1	Contents	3
1.1	Usage	3
1.2	Architecture	4
1.3	API	4

An account abstraction proposal which completely avoids the need for consensus-layer protocol changes. Instead of adding new protocol features and changing the bottom-layer transaction type, this proposal instead introduces a higher-layer pseudo-transaction object called a **UserOperation**.

Users send UserOperation objects into a separate mempool. A special class of actor called bundlers (either miners, or users that can send transactions to miners through a bundle marketplace) package up a set of these objects into a transaction making a `handleOps()` call to a special contract, and that transaction then gets included in a block.

This project is an implementation of [EIP-4337](#) Account Abstraction

This project provides:

- The core contracts and interfaces ([on github](#))
- Javascript client library
- Test suite for the contracts.

Check out the [Usage](#) section for further information, including how to [Installation](#) the project.

Note: The contracts of the project are audited.

The client library is under active development.

CONTENTS

1.1 Usage

1.1.1 Installation

To install locally, do:

```
git clone https://github.com/eth-infinity/account-abstraction.git
cd account-abstraction
yarn install
yarn test
```

To run a sample flow, calling locally the `EntryPoint.handleOps()` (note that for non-local network, you need to set `MNEMONIC_FILE` environment var to an account that has some eth)

```
yarn run runop [ --network hardhat|goerli ]
```

To run a sample operation against the goerli miner do

```
yarn run runop-goerli
```

1.1.2 Just using the Contracts:

```
yarn install @account-abstraction/contracts
```

From solidity code:

```
import "@account-abstraction/contracts/BaseWallet.sol";

contract MyContract is BaseWallet {
  ...
}
```

Just JSON artifacts:

```
artifact = require('@account-abstraction/contracts/artifacts/EntryPoint.json')
```

js/ts code:

```
const {EntryPoint__factory} = require("@account-abstraction/contracts/typechain/  
↳factories/EntryPoint__factory");  
...  
const entryPoint = EntryPoint__factory.connect(entryPointAddress, signer)
```

1.2 Architecture

1.2.1 Components

- EntryPoint contract
- Paymaster contract
- UserOperation
- Bundler
- Miner
- client library

1.3 API