Hardhat Verification Plugin

Hardhat is a full-featured development environment for contract compilation, deployment and verification. TheHardhat Etherscan plugin supports contract verification on BlockScout.

Get Started

1) Install Hardhat

If you are starting from scratch, create an npm project by going to an empty folder, runningnpm init , and following the instructions. Recommend npm 7 or higher.

Once your project is ready:

npm instructions

```

Copy npm install --save-dev hardhat

```

yarn instructions

```

Copy yarn add --dev hardhat

```

2) Create a project

Runnpx hardhat in your project folder and follow the instructions to create a(more info here).

3) Install plugin

Install thehardhat-etherscan plugin (requiresv3.0.0+).

npm

```

Copy npm install --save-dev @nomiclabs/hardhat-etherscan

```

yarn

```

Copy yarn add --dev @nomiclabs/hardhat-etherscan

```

4) Add plugin reference to config file

Add the following statement to yourhardhat.config.js .

```

Copy require("@nomiclabs/hardhat-etherscan");

```

If using TypeScript, add this to yourhardhat.config.ts. More info on using typescript with hardhat available here.

```

Copy import "@nomiclabs/hardhat-etherscan";

```

Config File

Your basicHardhat config file (hardhat.config.js orhardhat.config.ts) will be setup to support the network you are working on. In this example we use the Sokol test network and a.js file.

Here we add an RPC url without an API key, however some value is still required. You can use any arbitrary stringMore info .

If you prefer, you can migrate tohardhat-toolbox to use a plugin bundle.

```

Copy require("@nomiclabs/hardhat-waffle"); require("@nomiclabs/hardhat-etherscan"); require('hardhat-deploy');

let secret = require("./secret");

module.exports = { solidity: "0.8.9", networks: { sokol: { url: 'https://sokol.poa.network/', accounts: [secret.key], } }, etherscan: { // Your API key for Etherscan // Obtain one at https://etherscan.io/ apiKey: "abc" } };

```

Add an Unsupported Network

Some chains are not supported by the plugin (to check runnpx hardhat verify --list-networks)

If your chain is not in the list, you can add acustomChains object to the config file. It includes:

- chainID
-
    - Network chain ID
- apiURL
-
    - Block explorer API URL
- browserURL
-
    - Block explorer URL
-

Find an extensive list of ChainIDs athttps://chainlist.org/ . For example, if Sokol were not in the default list, this is how it would be added to the config file. Note the network name incustomChains must match the network name in theapiKey object.

```

Copy etherscan: { apiKey: { sokol: "abc" }, customChains: [ { network: "sokol", chainId: 77, urls: { apiURL: "https://blockscout.com/poa/sokol/api", browserURL: "https://blockscout.com/poa/sokol" } } ] }

```

Deploy and Verify

Deploy

```

Copy D:\hard_hat>npx hardhat run scripts\deploy.js --network sokol Contract deployed to: 0x8595e22825Ba499dB8C77C5c830c235D80f9C0fa

```

Verify

You can include constructor arguments with the verify task.

```

Copy npx hardhat verify --networkDEPLOYED_CONTRACT_ADDRESS "Constructor argument 1"

```

Sokol example (no constructors).

```

Copy D:\hard_hat>npxhardhatverify--networksokol0x8595e22825Ba499dB8C77C5c830c235D80f9C0fa Nothingtocompile
Compiling1filewith0.8.0 Successfullysubmittedsourcecodeforcontract
contracts/test.sol:Fooat0x8595e22825Ba499dB8C77C5c830c235D80f9C0fa forverification on Etherscan.
Waitingforverification result...

SuccessfullyverifiedcontractFooonEtherscan.
https://blockscout.com/poa/sokol/address/0x8595e22825Ba499dB8C77C5c830c235D80f9C0fa#code

```

Note the verify task will not be listed in the available tasks lists atnpx hardhat --config but should work as expected.

If not, check you have the minimum required version of the nomiclabs-hardhat-etherscan plugin (v3.0.0+) installed

Confirm Verification on BlockScout

Go to your BlockScout instance and paste the contract address into the search bar.

?

Scroll down to see verified status. A green checkmark ✅ means the contract is verified.

?

If your screen size is limited, you may need to click the 3 dots to view and click through to the contract.

?

Scroll down to see and interact with the contract code.

?

FAQ

I am using anOpenZeppelin upgrades plugin implementation and receive an error on proxy contract verification. What should I do? Although you receive an error, the contracts should be verified during the previous steps and you can ignore. Check in the explorer to make sure the contracts have been verified.

Resources

Learn more about plugin configs, troubleshooting etc. athttps://hardhat.org/plugins/nomiclabs-hardhat-etherscan.html

Last updated2 months ago