

# Blockscout smart-contract verification API

Blockscout also offers a contract verification API.

This is the preferred option for contract verification via API. However, you can also use RPC endpoints [more info is available here](#).

## License type

You can specify license type of the smart contract as string or number. For example for GNU General Public License v2.0 (GNU GPLv2) you could pass either 4 or "gnu\_gpl\_v2"

We are supporting such types of license as:

...

Copy 1. No License (None) 2. The Unlicense (Unlicense) 3. MIT License (MIT) 4. GNU General Public License v2.0 (GNU GPLv2) 5. GNU General Public License v3.0 (GNU GPLv3) 6. GNU Lesser General Public License v2.1 (GNU LGPLv2.1) 7. GNU Lesser General Public License v3.0 (GNU LGPLv3) 8. BSD 2-clause "Simplified" license (BSD-2-Clause) 9. BSD 3-clause "New" Or "Revised" license\* (BSD-3-Clause) 10. Mozilla Public License 2.0 (MPL-2.0) 11. Open Software License 3.0 (OSL-3.0) 12. Apache 2.0 (Apache-2.0) 13. GNU Affero General Public License (GNU AGPLv3) 14. Business Source License (BSL 1.1)

...

API license types:

...

Copy none unlicense mit gnu\_gpl\_v2 gnu\_gpl\_v3 gnu\_lgpl\_v2\_1 gnu\_lgpl\_v3 bsd\_2\_clause bsd\_3\_clause mpl\_2\_0 osl\_3\_0 apache\_2\_0 gnu\_agpl\_v3 bsl\_1\_1

...

## Verify smart contract

Use the appropriate Blockscout instance endpoint to verify if the smart contract microservice is enabled.

In the following examples we use <https://eth.blockscout.com> to query Ethereum.

?

0x contract addresses in POST example urls below should be replaced with your contract hash supplied on contract creation. Variables in the body are examples and should be replaced with your contract details.

## Flattened contract

For more information on parameters to pass, see the [flattened source code information on the Verifying a smart contract page](#).

?

## Via Standard JSON input

For more information on parameters to pass, see the [flattened source code information on the Verifying a smart contract page](#). 0x contract in POST example should be replaced with your contract hash.

?

## Via Sourcify Files

For more information on parameters to pass, see the [Contract Verification via Sourcify](#). 0x contract in POST example url should be replaced with your contract hash, as should your relevant variables.

?

## Multi-part Solidity files

?

## Vyper Contracts

?

Vyper Multi-part files

?

Vyper Standard JSON input

?

Last updated 3 months ago