Standards Track: Core

EIP-6780: SELFDESTRUCT only in same transaction

SELFDESTRUCT will recover all funds to the target but not delete the account, except when called in the same transaction as creation

Authors Guillaume Ballet (@gballet), Vitalik Buterin (@vbuterin), Dankrad Feist (@dankrad)

Created 2023-03-25

Requires EIP-2681, EIP-2929, EIP-3529

Table of Contents

Abstract

This EIP changes the functionality of the `SELFDESTRUCT` opcode. The new functionality will be only to send all Ether in the account to the target, except that the current behaviour is preserved when `SELFDESTRUCT` is called in the same transaction a contract was created.

Motivation

The `SELFDESTRUCT` opcode requires large changes to the state of an account, in particular removing all code and storage. This will not be possible in the future with Verkle trees: Each account will be stored in many different account keys, which will not be obviously connected to the root account.

This EIP implements this change. Applications that only use `SELFDESTRUCT` to retrieve funds will still work. Applications that only use `SELFDESTRUCT` in the same transaction as they created a contract will also continue to work without any changes.

Specification

The behaviour of `SELFDESTRUCT` is changed in the following way:

1. When `SELFDESTRUCT` is executed in a transaction that is not the same as the contract calling `SELFDESTRUCT` was created:
   - The current execution frame halts.
   - `SELFDESTRUCT` does not delete any data (including storage keys, code, or the account itself).
   - `SELFDESTRUCT` transfers the entire account balance to the target.
   - Note that if the target is the same as the contract calling `SELFDESTRUCT` there is no net change in balances. Unlike the prior specification, Ether will not be burnt in this case.
   - Note that no refund is given since EIP-3529.
   - Note that the rules of EIP-2929 regarding `SELFDESTRUCT` remain unchanged.
2. When `SELFDESTRUCT` is executed in the same transaction as the contract was created:
   - `SELFDESTRUCT` continues to behave as it did prior to this EIP, this includes the following actions
     - The current execution frame halts.
     - `SELFDESTRUCT` deletes data as previously specified.
     - `SELFDESTRUCT` transfers the entire account balance to the target

        - The account balance of the contact calling `SELFDESTRUCT` is set to `0`.
    - Note that if the target is the same as the contract calling `SELFDESTRUCT` that Ether will be burnt.
    - Note that no refund is given since EIP-3529.
    - Note that the rules of EIP-2929 regarding `SELFDESTRUCT` remain unchanged.

A contract is considered created at the beginning of a create transaction or when a CREATE series operation begins execution (CREATE, CREATE2, and other operations that deploy contracts in the future). If a balance exists at the contract's new address it is still considered to be a contract creation.

The `SELFDESTRUCT` opcode remains deprecated as specified in EIP-6049. Any use in newly deployed contracts is strongly discouraged even if this new behaviour is taken into account, and future changes to the EVM might further reduce the functionality of the opcode.

Rationale

Getting rid of the `SELFDESTRUCT` opcode has been considered in the past, and there are currently no strong reasons to use it. This EIP implements a behavior that will attempt to leave some common uses of `SELFDESTRUCT` working, while reducing the complexity of the change on EVM implementations that would come from contract versioning.

Handling the account creation and contract creation as two distinct and possibly separate events is needed for use cases such as counterfactual accounts. By allowing the `SELFDESTRUCT` to delete the account at contract creation time it will not result in stubs of counterfactually instantiated contracts that never had any on-chain state other than a balance prior to the contract creation. These accounts would never have any storage and thus the trie updates to delete the account would be limited to the account node, which is the same impact a regular transfer of ether would have.

Backwards Compatibility

This EIP requires a hard fork, since it modifies consensus rules.

Contracts that depended on re-deploying contracts at the same address using `CREATE2` (after a `SELFDESTRUCT`) will no longer function properly if the created contract does not call `SELFDESTRUCT` within the same transaction.

Previously it was possible to burn ether by calling `SELFDESTRUCT` targeting the executing contract as the beneficiary. If the contract existed prior to the transaction the ether will not be burned. If the contract was newly created in the transaction the ether will be burned, as before.

Test Cases

Test cases for this EIP can be found in the Execution Spec Tests suite `eip6780_selfdestruct`.

Security Considerations

The following applications of `SELFDESTRUCT` will be broken and applications that use it in this way are not safe anymore:

1. Where `CREATE2` is used to redeploy a contract in the same place in order to make a contract upgradable. This is not supported anymore and ERC-2535 or other types of proxy contracts should be used instead.

2. Where a contract depended on burning Ether via a `SELFDESTRUCT` with the contract as beneficiary, in a contract not created within the same transaction.

Copyright

Copyright and related rights waived via CC0.

Citation

Please cite this document as:

Guillaume Ballet (@gballet), Vitalik Buterin (@vbuterin), Dankrad Feist (@dankrad), "EIP-6780: SELFDESTRUCT only in same transaction," *Ethereum Improvement Proposals*, no. 6780, March 2023. [Online serial]. Available: https://eips.ethereum.org/EIPS/eip-6780.