CRYPTO UNDER THE HOOD

CAHILLNXT

Deep Dive Into Tornado Cash

The Nuances of Immutability and its Legal Implications

January 30, 2025

The recent decision by the U.S. Court of Appeals for the Fifth Circuit in *Van Loon, et al.* v. *Dep't of the Treasury*, 122 F.4th 549 (2024) held that the U.S. Department of the Treasury's Office of Foreign Assets Control (OFAC) exceeded its statutory authority by sanctioning the suite of smart contracts (*i.e.*, computer code) comprising Tornado Cash—a decentralized, open-source privacy protocol that facilitates anonymous cryptocurrency transactions—because such smart contracts are not "property" subject to the sanctions jurisdiction asserted by OFAC within the meaning of the International Emergency Economic Powers Act (IEEPA). A central argument in the decision was the perceived "immutability" of the Tornado Cash smart contracts, with the court emphasizing that "once a smart contract becomes immutable, no one can reclaim control over it." However, a closer technical examination reveals that the notion of immutability is much more nuanced.

This article aims to provide a detailed technical overview of Tornado Cash and dissect the nuances of its smart contract architecture to demonstrate the importance of a technical understanding to avoid future pitfalls.

Case Background

In 2022, OFAC added a number of Ethereum addresses associated with the Tornado Cash protocol to the Specially Designated Nationals and Blocked Persons List (SDN List). Once a party or property is identified on the SDN List, all U.S. persons or persons within the U.S. are prohibited from engaging with that party or property, and any violations will be subject to strict liability criminal penalties.

Six individuals who used Tornado Cash to enhance their privacy for legitimate transactions challenged this designation in court. While the district court initially upheld the OFAC designation, the Fifth Circuit reversed, holding that immutable smart contracts, like those used by the Tornado Cash protocol, are not "property" within the meaning of the IEEPA because they are not capable of being owned, controlled, or altered by any individual or entity.

What is Tornado Cash?

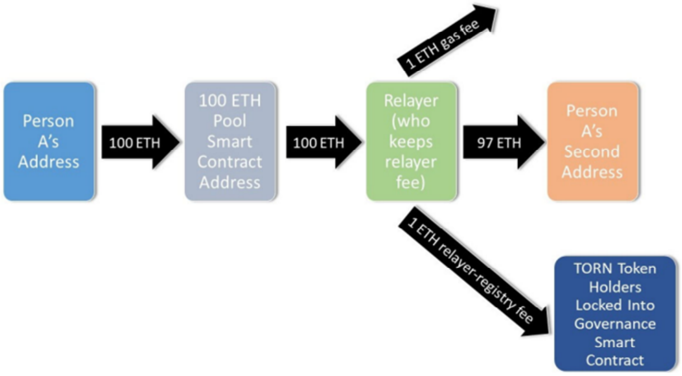
"Tornado Cash" is a blockchain-based protocol consisting of a suite of smart contracts deployed to the Ethereum blockchain network. Its primary function is to delink the origin and destination of crypto asset transactions, enhancing user privacy through a process known as "mixing."

How Does Tornado Cash Work?

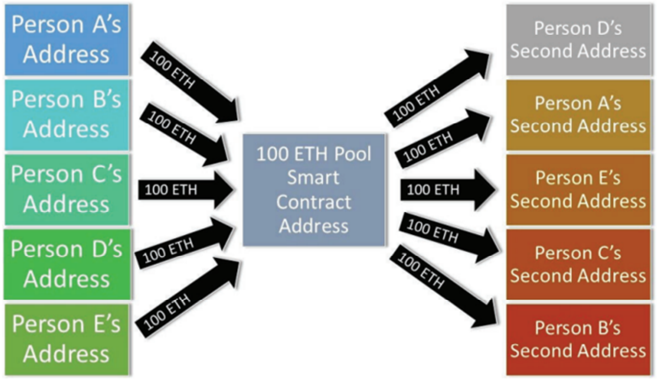
Here are the three primary functions of Tornado Cash:

- **Deposit:** Users execute a transaction sending a specific amount of a crypto asset (*e.g.*, ETH) from a given blockchain address to a Tornado Cash smart contract address. This transaction is associated with a unique "note" of the amount of assets transferred.
- **Mixing:** The transferred crypto assets are pooled with other transfers of the same asset sent to the smart contract. "Mixing" occurs after a protocol-determined number of transfers have occurred.
- **Withdrawal:** Users can then execute a second transaction to request the transfer of the same amount of the initially transferred asset (minus a portion of the transferred asset considered to be a fee) to a different address from the original sending address. Crucially, the assets sent from the Tornado Cash address are not directly traceable to the original deposit due to the mixing process.

The illustrative graphs drawn by the court in *Van Loon* are shown as below:


Person A's Address
100 ETH
100 ETH Pool Smart Contract Address
100 ETH
Relayer (who keeps relayer fee)
1 ETH gas fee
97 ETH
Person A's Second Address
1 ETH relayer-registry fee
TORN Token Holders Locked Into Governance Smart Contract

Van Loon at 559.


Person A's Address
Person B's Address
Person C's Address
Person D's Address
Person E's Address
100 ETH
100 ETH
100 ETH
100 ETH
100 ETH
100 ETH Pool Smart Contract Address
100 ETH
100 ETH
100 ETH
100 ETH
100 ETH
Person D's Second Address
Person A's Second Address
Person E's Second Address
Person C's Second Address
Person B's Second Address

Van Loon at 557.

Deposit

- **Choosing a Pool:** Users select a specific Tornado Cash smart contract address (or "pool") based on the type of crypto asset and the amount. Each pool accepts a fixed amount of a particular crypto asset (*e.g.*, 1 ETH, 100 DAI).

- **Generating a Secret:** When transferring to the relevant pool, the user's wallet generates a secret – *i.e.*, a random piece of data. This secret is crucial for withdrawing the user's assets later.

- **Creating a Commitment:** The secret is then used in the user's wallet to create a unique cryptographic hash (or "commitment"), which is essentially a "fingerprint" of the secret. This commitment is recorded on the Ethereum network, proving the user's deposit without revealing the secret itself.

- **Sending Funds to the Pool:** The user then sends the chosen amount of crypto assets to the pool, along with the commitment. The Tornado Cash smart contract stores this commitment in a Merkle tree, a cryptographic data structure that efficiently verifies the commitment's inclusion. This transaction is visible on the blockchain, but the link between the user's original sending address and the deposit is broken as the funds are "mixed" in the pool.

Mixing

- **Anonymity Sets:** The pool is as an anonymity set of select crypto assets, where the assets of every user in a given pool are mixed together. This makes it functionally impossible to trace the origin or destination of any particular transaction.

Withdrawal

- **Initiating the Withdrawal:**
    - **New Address:** The user starts by generating a new Ethereum address. This address will be used to receive the withdrawn funds, ensuring no direct link on the blockchain to its original deposit address.
    - **Proof Generation:** The user's digital asset wallet uses the secret, the nullifier of the deposit, which was calculated during the deposit by the user's wallet, and the Merkle root (representing the aggregated Hash values within a Merkle tree) of the current anonymity set to generate a zero-knowledge proof (a protocol designed to verify a particular statement such as "My Ethereum account deposited 100 ETH to Tornado Cash" without revealing sensitive information such as the amount deposited or the depositing address). This proof demonstrates that:
        - The user knows the secret corresponding to a valid commitment in the anonymity set; and
        - The nullifier has not been used before, preventing double-spending.

    - **Relayer (Optional):** A relayer is a third-party service that submits the withdrawal transaction on behalf of the user, obscuring the user's IP address and other identifying information, further enhancing privacy.

- **Interacting with the Tornado Cash Smart Contract:**
    - **Submitting the Proof:** The user (or the relayer, if used) submits the generated zero-knowledge proof and the new withdrawal address to the Tornado Cash smart contract in a withdrawal transaction.
    - **Verification:** The smart contract executes a function that verifies the zero-knowledge proof using the stored verifier key. This confirms that the proof is valid and the user is entitled to withdraw the funds.
    - **Nullifier Check:** The smart contract checks if the nullifier provided in the proof has already been used. If it has, the withdrawal is rejected to prevent double-spending.
    - **Merkle Tree Update:** If the proof is valid and the nullifier is unused, the smart contract updates the Merkle tree of the anonymity set by adding the nullifier. This marks the deposit as "spent."
    - **Sending to the New Address:** Finally, the specified amount of the relevant crypto asset is sent from the Tornado Cash pool to the user's new withdrawal address.

Zero-Knowledge Proof

Tornado Cash uses a specific type of zero-knowledge proof called a "zk-SNARK" which can be verified by a smart contract on the Ethereum network. zk-SNARK proofs are derived from a "circuit," a specialized cryptographic computer program, to verify transactions privately. The circuit is a crucial component of its anonymity mechanism. More specifically:

- **Circuit Inputs**: The circuit takes several inputs, both public and private (anonymous):
    - Public
        - The commitment generated using the secret;
        - The nullifier (a unique value generated during withdrawal to prevent double-spending); and
        - The Merkle root of the current state of the anonymity set.
    - Private
        - The user's secret (which is never revealed);
        - The commitment's "blinding factor"; and
        - The Merkle proof of non-inclusion of the nullifier.

- **Circuit Computation**: The circuit performs a series of cryptographic operations using these inputs to verify the following:
    - The user requesting the withdrawal knows the secret corresponding to the provided commitment;
    - The commitment is part of the current Merkle tree, proving the relevant amount of funds were deposited; and
    - The nullifier hasn't been used before through a proof of non-inclusion, preventing double-spending.
- **Output**: After several cryptographic steps, the circuit produces a zk-SNARK proof, which is a small piece of data that can be verified directly in a smart contract on the Ethereum network. The output proves only that the user that sent the proof had access to the secret data about a corresponding deposit transaction, and nothing more.

The Nuances of Immutability

Generally speaking, smart contracts comprising the Tornado Cash protocol are essentially lines of computer code that, when "called," automate the mixing and proof verification processes. As emphasized by the Fifth Circuit Court of Appeals, the concept of code "immutability" suggests that, once these lines of code have been deployed to the blockchain network, "no one can reclaim control over it." The court's analysis of the IEEPA's scope hinged on how this immutability affected property ownership, noting that the statute only addresses "property," which it reasoned must be "capable of being owned." As the court explained, "when someone has a property interest, he or she typically has the 'rights of possession and control.'" The court went on to conclude that such rights were nonexistent with Tornado Cash's immutable smart contracts since they "became self-executing and could no longer be altered, removed, or controlled" after deployment—placing them outside of the IEEPA's purview.

However, there are at least three aspects relevant to true immutability:

- **Upgradability**: While the core logic of a smart contract may be immutable, many modern smart contracts incorporate mechanisms for upgradability. This means that the contract's "owners" (usually the person or persons that developed the code) retain a "private key" that allow them to deploy new versions of the code with modified functionality to the same blockchain address. While the original code remains visible on the blockchain, the active contract code can be switched to the upgraded version without changing the smart contract's network address.
- **Proxy Contracts**: Tornado Cash utilizes a "proxy contract" architecture. In this model, the user interacts with a proxy contract that forwards calls to the underlying "implementation contract". This architecture allows for upgrades to the implementation contract without affecting the user's interaction point (*i.e.*, the proxy contract).
- **Governance**: Some smart contracts include governance mechanisms that allow token holders or a designated group to vote on changes to the contract's parameters or even its basic logic.

In the case of Tornado Cash, the underlying smart contracts (including the implementation contracts) are not upgradable. The Tornado Cash smart contract system specifies "governance" as an Ethereum-based address that can invoke certain restricted functions, such as rescuing tokens sent by accident to the smart contract pool. However, this governance feature cannot make changes to the logic of the underlying smart contract code. The governance address is controlled by a "decentralized autonomous organization" (DAO) associated with Tornado Cash comprised of TORN token holders. Following a governance process, this address can perform operations that all other users cannot. While the TORN token has been delisted from most major trading platforms, it is still available for purchase via peer-to-peer trading.

As long as governance is not set to an address that cannot initiate transactions on the Ethereum network, such as the "Zero Address" (where all bytes are set to zero, signifying a null address), then the Tornado Cash smart contracts are not strictly speaking logically "immutable" (*i.e.*, elements of the contracts can still be altered). However, since the governance address itself is specified in the Tornado cash smart contract code, that governance address can no longer be changed, meaning that the token holders comprising the Tornado Cash DAO (and only these token holders) will always have privileged access to certain Tornado Cash smart contract functions.

The "Immutability" Argument

The potential for upgrades, the use of proxy contracts, and the presence of governance mechanisms introduce a degree of flexibility when assessing the immutability of any smart contract. Each of these aspects can be leveraged to modify the behavior of a protocol.

Before relying on any argument of immutability, one should proceed with caution and a deep understanding of how the relevant smart contracts work. This is critical when determining whether code deployed to a blockchain network is truly immutable, especially in high-risk areas such as sanctions. Users should also have a deep understanding of the underlying technology to assess any risk arising from interactions with smart contract systems that are not fully immutable.

While the *Van Loon* decision raises important questions and answers some (particularly about the reach of IEEPA and its application to blockchain-based activities), it is crucial to ground these discussions in a sound technical understanding of the underlying technology. The "immutability" argument, while appealing in its simplicity, may not fully capture the complexities of various smart contract architectures and their potential for evolution. We note that the *Van Loon* decision may still be appealed by the Treasury Department, and a parallel case is pending in the Eleventh Circuit. *See Coin Center, et al.* v. *Secretary, U.S. Department of the Treasury, et al.*, No. 23-13698 (11th Cir. filed Nov. 7, 2023).

* * *

If you have any questions about the issues addressed in this publication, please reach out to the CahillNXT team at CahillNXT@cahill.com. To learn more about CahillNXT, the Digital Assets and Emerging Technology practice at Cahill Gordon & Reindel LLP, click here.