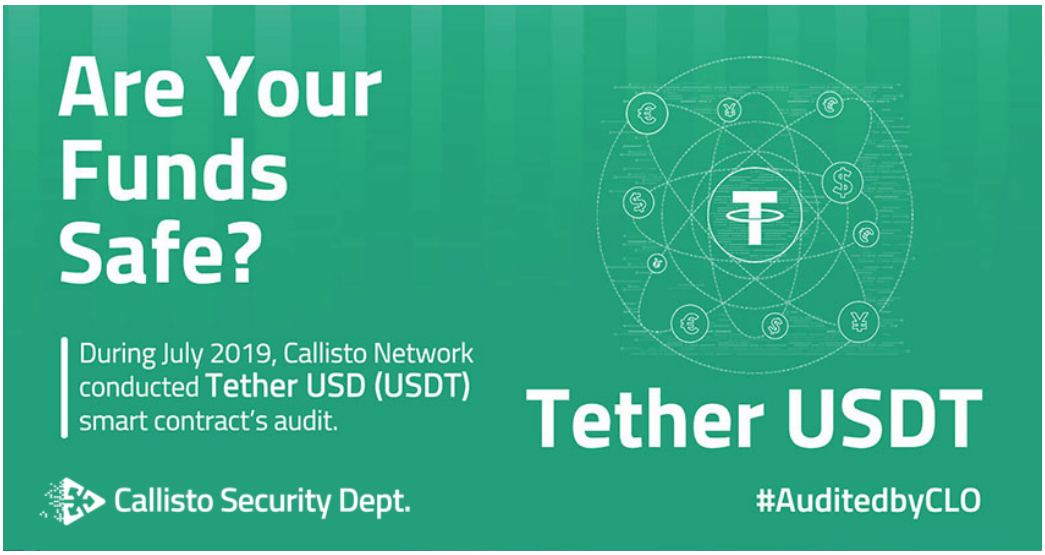
Tether Token (USDT) Security Audit



Tether Token (USDT) security audit, conducted by the Callisto Network Security Department in July 2019.

Tether Token (USDT) Specificities

Audit Request

Audit Top 200 CoinMarketCap tokens.

```
Symbol      : USDT
Name        : Tether USD
```

Deployed at:

https://etherscan.io/address/0xdac17f958d2ee523a2206206994597c13d831ec7#contracts (https://etherscan.io/address/0xdac17f958d2ee523a2206206994597c13d831ec7#contracts)

Source Code:

https://etherscan.io/address/0xdac17f958d2ee523a2206206994597c13d831ec7#contracts (https://etherscan.io/address/0xdac17f958d2ee523a2206206994597c13d831ec7#contracts)

Disclosure policy

Public.

Platform:

ETH.

Number of lines:

252.

Tether Token (USDT) Smart Contract Security Audit Report

Are Your Funds Safe?

---

1. In scope

- USDT solidity contract (https://etherscan.io/address/0xdac17f958d2ee523a2206206994597c13d831ec7#contracts)

2. Findings

In total, **7 issues** were reported including:

- 3 low severity issues.
- 4 owner privileges (the ability of an owner to manipulate contract, may be risky for investors).

No critical security issues were found.

2.1. Owner Privileges

Severity: Owner Privileges.

Description:

1. Pause/unpause `transfer` (https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L340) and

`transferFrom` (https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L350).

2. Blacklist users addresses individualy from using `transfer` (https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L340) and `transferFrom` (https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L350) through `addBlackList` (https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L281) and `removeBlackList` (https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L286). And burn blacklisted users assets (please note that the assets are supposed to be backed in a 1:1 ratio) using `destroyBlackFunds` (https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L291) function.
3. A maximal fee percentage of 20/10000 can be applied per transaction with a maximal value of 50 USDT, check here (https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L429).
4. Owner can upgrade contract using `deprecate` (https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L387) and implement any logic in the new contract. And even if the new contract will be audited, at any time possible to change the address of the new contract again to not audited and insecure.

2.2. Transfert to Address (0)

Severity: low.

Description:

- `transfer` and `transferFrom` allow the destination address to be equal to zero, meaning that users fund can be lost if sent to it by mistake.

Code snippet:

```
function transfer(address _to, uint _value) public onlyPayloadSize(2 * 32) {
    uint fee = (_value.mul(basisPointsRate)).div(10000);
    if (fee > maximumFee) {
        fee = maximumFee;
    }
    uint sendAmount = _value.sub(fee);
    balances[msg.sender] = balances[msg.sender].sub(_value);
    balances[_to] = balances[_to].add(sendAmount);
    if (fee > 0) {
        balances[owner] = balances[owner].add(fee);
        Transfer(msg.sender, owner, fee);
    }
    Transfer(msg.sender, _to, sendAmount);
}
```

```
function transferFrom(address _from, address _to, uint _value) public onlyPayloadSize(3 * 32) {
        var _allowance = allowed[_from][msg.sender];

        uint fee = (_value.mul(basisPointsRate)).div(10000);
        if (fee > maximumFee) {
            fee = maximumFee;
        }
        if (_allowance < MAX_UINT) {
            allowed[_from][msg.sender] = _allowance.sub(_value);
        }
        uint sendAmount = _value.sub(fee);
        balances[_from] = balances[_from].sub(_value);
        balances[_to] = balances[_to].add(sendAmount);
        if (fee > 0) {
            balances[owner] = balances[owner].add(fee);
            Transfer(_from, owner, fee);
        }
        Transfer(_from, _to, sendAmount);
    }
```

2.3. Not Emitted Transfert Event

Severity: low.

Description:

When issuing or redeeming tokens a transfer event should be emitted back and forth to address(0). ERC20 standard (https://eips.ethereum.org/EIPS/eip-20): “A token contract which creates new tokens SHOULD trigger a Transfer event with the _from address set to 0x0 when tokens are created”. the same can be deducted when redeeming or burning tokens.

Same issue is applicable for `TetherToken` constructor (https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L330).

Code snippet:

https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L406 (https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L406)

https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L420 (https://gist.github.com/RideSolo/24c79eb34b565ade477ec89c2af49a5b#file-usdt-sol-L420)

2.4. Known vulnerabilities of ERC-20 token

Severity: low.

Description:

- It is possible to double withdrawal attack. More details here (https://docs.google.com/document/d/1YLPtQxZu1UAvO9cZ1O2RPXBbT0mooh4DYKjA_jp-RLM/edit)
- Lack of transaction handling mechanism issue. WARNING! (https://gist.github.com/Dexaran/ddb3e89fe64bf2e06ed15fbd5679bd20) This is a very common issue and it already caused millions of dollars losses for lots of token users! More details here. (https://docs.google.com/document/d/1Feh5sP6oQL1-1NHi-X1dbgT3ch2WdhbXRevDN681Jv4/edit)

Recommendation:

> **Add the following code to the `transfer(_to address, ...)` function:**
>
> ```
> require( _to != address(this) );
> ```

3. Conclusion

The audited smart contract can be deployed. Only low severity issues were found during the audit.

4. Revealing audit reports

- https://gist.github.com/yuriy77k/49b74a164bccac9b2554de9b25ffae8b (https://gist.github.com/yuriy77k/49b74a164bccac9b2554de9b25ffae8b)
- https://gist.github.com/yuriy77k/476b9556f4895d32867890af4e4199ba (https://gist.github.com/yuriy77k/476b9556f4895d32867890af4e4199ba)
- https://gist.github.com/yuriy77k/d75e9365b6ec7eefd46a237d09e673bc (https://gist.github.com/yuriy77k/d75e9365b6ec7eefd46a237d09e673bc)

Appendix

Smart Contract Audits by Callisto Network. (https://callisto.network/smart-contract-audit/)

Miscellaneous

Why Audit Smart Contracts? (https://callisto.network/why-audit-smart-contracts/)

Our Most Popular Audit Reports. (https://callisto.network/security-audits/)

Trust the Blockchain, Audit the Smart Contracts.

Follow Callisto's Security Department on Twitter (https://twitter.com/Callisto_Audits) to get our latest news and updates!

Published on **September 7, 2020**

(https://callisto.network/tether-token-usdt-security-audit/)

Security Audits (https://callisto.network/tag/security-audits/)

‹ Previous post (https://callisto.network/passive-income-with-crypto-guarda-wallet/) Next post ›

Callisto Network LTD

71-75 Shelton Street
London, Greater London
United Kingdom, WC2H 9JQ

Join Our Community

(https://t.me/CallistoNet) (https://twitter.com/CallistoSupport) (http://reddit.com/r/CallistoCrypto) (https://www.youtube.com/channel/UC1WMae32v_eJ8qOtLQ

(https://www.instagram.com/callisto.network/) (https://www.facebook.com/callistonetwork) (https://www.linkedin.com/company/callisto-network/) (https://t.co/DAWunSR1tm)

Resources

FAQ (https://callisto.network/faq/)

Timeline (https://callisto.network/timeline/)

Airdrop (https://callisto.network/callisto-airdrop/)

Community Guidelines (https://callisto.network/community-guidelines/)

Callisto

Partners (https://callisto.network/partners/)

Our GitHub repositories (https://github.com/EthereumCommonwealth)

Media Kit (https://github.com/EthereumCommonwealth/Callisto-Media-Kit)

Contact us (https://callisto.network/contact-us/)

Want to sell your CLO coins OTC? (mailto:vladimir.vencalek@invictussolutions.cz)