# Smart Contract Security Audit Report

Request Date: 21.10.2022
Last Revision: 23.11.2022
Version: 2


SEDAN - SERVICES AND DATA MANAGEMENT GROUP
SnT - Interdisciplinary Centre for Security, Reliability and Trust
University of Luxembourg

JFK Building - 29, avenue J. F. Kennedy, L-1855 Luxembourg

## Lead Auditor

Radu State
<radu.state@uni.lu>

## Co-Auditors

Antonio Ken Iannillo
<antonioken.iannillo@uni.lu>

Bahareh Parhizkari
<bahareh.parhizkari@uni.lu>

# Disclaimer

Information security threats are continually changing, with new vulnerabilities discovered on a daily basis, and no application can ever be 100% secure no matter how much security testing is conducted. This report cannot and does not protect against personal or business loss as the result of use of the applications or systems described. SEDAN offers no warranties, representations or legal certifications concerning the applications or systems it tests. All software includes defects: nothing in this document is intended to represent or warrant that security testing was complete and without error, nor does this document represent or warrant that the application tested is suitable to task, free of other defects than reported, fully compliant with any industry standards, or fully compatible with any operating system, hardware, or other application. By using this information you agree that SEDAN be held harmless in any event. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes without SEDAN's prior written consent. Notwithstanding above, this report may be disclosed to governmental or other regulatory authorities and published on the official website of the project. This report was conducted in the context of a joint research project between VNX and the Interdisciplinary Centre for Security, Reliability and Trust (SnT).

# Contents

# 1 Executive Summary

This report summarizes the results of our smart contract security audit that was requested by VNX. We run a wide range of state-of-the-art security analysis tools to assess the security of the smart contracts of VNX.

## 1.1 Version 2

This audit follows another audit from SEDAN on 12/11/2022. We run again the wide range of state-of-the-art security analysis tools to assess the security of the smart contracts of VNX.

# 2 Version History

| Version | Date | Comments |
|---|---|---|
| 1 | 14.11.2022 | First round of the second version of the audit report |
| 2 | 23.11.2022 | Second round of the second version of the audit report |

# 3 Audit Scope

The scope of this audit is limited to a selected subset of smart contracts that are related to the Stablecoin Token project of VNX, as requested by VNX.

## 3.1 Files Covered

The audit covers all the files that are listed below, located in the "`contracts`" folder of the provided repository.

### 3.1.1 Version 1

| | |
|---|---|
| **Repository** | `https://gitlab.com/vnx/ethereum-contracts` |
| **Commit** | `80866ec9` |
| **SHA-1 of the downloaded zip file** | `a8ff90dcea0340c7b52856e5c7b118401115468f` |

Files' origin information

| **Filename** | **SHA-1 Hash** |
|---|---|
| VNXCToken.sol | `5086b4c8e01b33d427ef9d91e9435d9b9a3b1678` |
| Manager.sol | `02971f4576bd2dffbf8aaad1bb141e619b446b39` |

Files covered in version 1

### 3.1.2 Version 2

| | |
|---|---|
| **Repository** | `https://gitlab.com/vnx/ethereum-contracts` |
| **Commit** | `887f5fff` |
| **SHA-1 of the downloaded zip file** | `6f47e35a2d7244072b22197bbcd52a0a6d4fa1b6` |

Files' origin information

| Filename | SHA-1 Hash |
| --- | --- |
| VNXCToken.sol | `90020f7e1c39404b79e13527d6654d9c28467d66` |
| Manager.sol | `02971f4576bd2dffbf8aaad1bb141e619b446b39` |
| ProxyAdmin.sol | `ba8a30006ed6ae0ed3bda8c1089513ae93e9313e` |
| AnyTransferProvider.sol | `6a9ed6d1024445086b30c377ecf72478b359f41d` |

Files covered in version 2

# 4 Methodology

Our audit is composed of an automated security assessment using a collection of state-of-the-art smart contract analysis tools.

## 4.1 Tools

The following table lists the security analysis tools that were used during our audit:

| Toolname | Version | Description |
|---|---|---|
| Solc | 0.8.17 | Solc is the standard Solidity compiler maintained by the Ethereum Foundation. The compiler reports compilation warnings and errors. |
| Mythril | 0.23.10 | Mythril is a mature symbolic execution tool maintained by ConSensys Diligence to detect smart contract vulnerabilities. |
| Slither | 0.9.1 | Slither is a static analysis tool to find vulnerabilities within smart contracts that have been written in Solidity. |

## 4.2 Severity

We mark each finding with one of the following three severity labels:

| Severity | Description |
|---|---|
| Minor | Minor issues are subjective in nature. They are typically suggestions around best practices or readability. Code maintainers should use their own judgment as to whether to address such issues. |
| Major | Major issues are security vulnerabilities that may not be directly exploitable or may require certain conditions in order to be exploited. All major issues should be addressed. |
| Critical | Critical issues are directly exploitable security vulnerabilities that need to be fixed. |

## 4.3 Status

We mark each finding with one of the following three status labels:

| Status | Description |
|---|---|
| ✓ | This means that the issue has been fixed. |
| ✗ | This means that the issue has not been fixed. |
| - | This means that our recommendation is to ignore the issue. |

# 5 Findings

The following table summarizes all our findings:

| Filename | Finding | Severity | Status |
|---|---|---|---|
| `Manager.sol:35` | External Call To User-Supplied Address | Minor | - |
| `Manager.sol:35` | Multiple Calls in a Single Transaction | Minor | - |
| `Manager.sol:2, IRBAC.sol:2` | Old versions allowed | Minor | ✗ |
| `VNXCToken.sol:27:1` | Contract Code Size exceeds 24576 bytes | Major | ✓ |
| `VNXCToken.sol:91` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:101` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:112` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:122` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:162` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:163` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:164` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:165` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:202` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:306` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:357` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:367` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:394` | Conformance to Solidity naming conventions | Minor | - |
| `VNXCToken.sol:27-394` | Unused State Variable | Minor | - |
| `VNXCToken.sol:306-317` | Reentrancy | Minor | ✗ |
| `VNXCToken.sol:202-222` | Reentrancy | Minor | ✗ |
| `AnyTransferProvider.sol` | Unused function parameter | Minor | - |

## 5.1 Manager.sol

### 5.1.1 Solidity Compiler

No issues were detected.

### 5.1.2 Mythril

| **External Call To User-Supplied Address** | |
|---|---|
| `to ignore` | |
| `Manager.sol:35` | Minor |
| **Description**<br>A call to a user-supplied address is executed.<br>An external message call to an address specified by the caller is executed. Note that the callee account might contain arbitrary code and could re-enter any function within this contract. Reentering the contract in an intermediate state may lead to unexpected behaviour. | |
| **Recommendation**<br>As there exists no other functions after this call, the current code is safe. Just make sure not to add any code after this function call in the future. Ignore it. | |

| **Multiple Calls in a Single Transaction** | |
|---|---|
| `to ignore` | |
| `Manager.sol:35` | Minor |
| **Description**<br>This call is executed following another call within the same transaction. It is possible that the call never gets executed if a prior call fails permanently.<br>This is the expected behaviour. Ignore it. | |
| **Recommendation**<br>If possible, refactor the code such that each transaction only executes one external call or make sure that all callees can be trusted (i.e. they're part of your own codebase). | |

### 5.1.3 Slither

| **Old versions allowed** | |
|---|---|
| `Not solved yet` | |
| `Manager.sol:2, IRBAC.sol:2` | Minor |
| **Description**<br>Pragma version^0.8.0 allows old versions. | |
| **Recommendation**<br>Change the pragma version to 0.8.17 so that it's in line with the other contracts.<br>Using an old version prevents access to new Solidity security checks. | |

## 5.2 VNXCToken.sol

### 5.2.1 Solidity Compiler

| **Contract Code Size exceeds 24576 bytes.** | |
|---|---|
| `Solved in version 2` | |
| `VNXCToken.sol:27:1` | Major |
| **Description**<br>Contract code size is 24727 bytes and exceeds 24576 bytes (a limit introduced in Spurious Dragon). This contract may not be deployable on Mainnet. | |
| **Recommendation**<br>Check if contract is deployable by deploying the contract on a local geth node with the latest hard fork. In case the contract is not deployable, consider enabling the optimizer (with a low ”runs” value!), turning off revert strings, or using libraries. | |

### 5.2.2 Mythril

No issues were detected.

### 5.2.3 Slither

| **Conformance to Solidity naming conventions** | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:91` | Minor |
| **Description**<br>Parameter `_account` in isFrozen is not in mixedCase. | |
| **Recommendation**<br>Ignore, since it is already following the Solidity naming conventions.<br>It’s a false positive from Slither. | |

| Conformance to Solidity naming conventions | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:101` | Minor |
| **Description**<br>Parameter `_newMinterRoleId` in setMinterRole is not in mixedCase. | |
| **Recommendation**<br>Ignore, since it is already following the Solidity naming conventions.<br>It's a false positive from Slither. | |

| Conformance to Solidity naming conventions | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:112` | Minor |
| **Description**<br>Parameter `_newAssetProtectionRoleId` in setAssetProtectionRole is not in mixedCase. | |
| **Recommendation**<br>Ignore, since it is already following the Solidity naming conventions.<br>It's a false positive from Slither. | |

| Conformance to Solidity naming conventions | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:122` | Minor |
| **Description**<br>Parameter `_newProvider` in changeTransferProvider is not in mixedCase. | |
| **Recommendation**<br>Ignore, since it is already following the Solidity naming conventions.<br>It's a false positive from Slither. | |

| Conformance to Solidity naming conventions | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:162` | Minor |

**Description**

Parameter `_tokenName` in initialize is not in mixedCase.

Parameter `_tokenSymbol` in initialize is not in mixedCase.

**Recommendation**

Ignore, since it is already following the Solidity naming conventions.
It's a false positive from Slither.

| Conformance to Solidity naming conventions | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:163` | Minor |

**Description**

Parameter `_tokenCurrency` in initialize is not in mixedCase.

Parameter `_assetProtectionRoleId` in initialize is not in mixedCase.

**Recommendation**

Ignore, since it is already following the Solidity naming conventions.
It's a false positive from Slither.

| Conformance to Solidity naming conventions | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:164` | Minor |

**Description**

Parameter `_minterRoleId` in initialize is not in mixedCase.

Parameter `_newOwner` in initialize is not in mixedCase.

Parameter `_rbac` in initialize is not in mixedCase.

**Recommendation**

Ignore, since it is already following the Solidity naming conventions.
It's a false positive from Slither.

| Conformance to Solidity naming conventions | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:165` | Minor |
| **Description** | |
| Parameter `_initTransferProvider` in initialize is not in mixedCase. | |
| **Recommendation** | |
| Ignore, since it is already following the Solidity naming conventions.<br>It's a false positive from Slither. | |

| Conformance to Solidity naming conventions | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:202` | Minor |
| **Description** | |
| Parameter `_to` in mint is not in mixedCase.<br>Parameter `_amount` in mint is not in mixedCase. | |
| **Recommendation** | |
| Ignore, since it is already following the Solidity naming conventions.<br>It's a false positive from Slither. | |

| Conformance to Solidity naming conventions | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:306` | Minor |
| **Description** | |
| Parameter `_from` in burn is not in mixedCase.<br>Parameter `_amount` in burn is not in mixedCase. | |
| **Recommendation** | |
| Ignore, since it is already following the Solidity naming conventions.<br>It's a false positive from Slither. | |

| Conformance to Solidity naming conventions | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:357` | Minor |

**Description**

Parameter `_addr` in freeze is not in mixedCase.

**Recommendation**

Ignore, since it is already following the Solidity naming conventions.
It's a false positive from Slither.

| Conformance to Solidity naming conventions | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:367` | Minor |

**Description**

Parameter `_addr` in unFreeze is not in mixedCase.

**Recommendation**

Ignore, since it is already following the Solidity naming conventions.
It's a false positive from Slither.

| Conformance to Solidity naming conventions | |
|---|---|
| `to ignore` | |
| `VNXCToken.sol:394` | Minor |

**Description**

Parameter `_spender` in approve is not in mixedCase.

Parameter `_value` in approve is not in mixedCase.

**Recommendation**

Ignore, since it is already following the Solidity naming conventions.
It's a false positive from Slither.

| Unused State Variable | |
|---|---|
| to ignore | |
| VNXCToken.sol:27-394 | Minor |

**Description**

State variable `ERC20PermitUpgradeable.__gap` is never used.
State variable `ERC20PermitUpgradeable._PERMIT_TYPEHASH_DEPRECATED_SLOT` is never used.

**Recommendation**

Ignore since these variables are supposed not to be used.

| Reentrancy | |
|---|---|
| Ensure trust | |
| VNXCToken.sol:306-317 | Minor |

**Description**

Reentrancy in `burn`

```
_burn(_from,_amount)  (contracts/VNXCToken.sol:314)
    require(bool,string)(transferProvider.approveTransfer(
        from,to,amount,msg.sender),Declined by Transfer
        Provider!) (contracts/VNXCToken.sol#191)
```

**Recommendation**

Ensure that the `transferProvider` is trusted (i.e. it's part of your own codebase).

| Reentrancy | |
|---|---|
| Ensure trust | |
| VNXCToken.sol:202-222 | Minor |

**Description**

Reentrancy in `mint`

```
_mint(_to,_amount) (contracts/VNXCToken.sol#218)
    require(bool,string)(transferProvider.approveTransfer(
        from,to,amount,msg.sender),Declined by Transfer
        Provider!) (contracts/VNXCToken.sol#191)
```

**Recommendation**

Ensure that the `transferProvider` is trusted (i.e. it's part of your own codebase).

## 5.3 AnyTransferProvider.sol

### 5.3.1 Solidity Compiler

| **Unused function parameter** | |
|---|---|
| `To ignore` | |
| `AnyTransferProvider.sol:12:69/21:29/43/55` | Minor |

**Description**

Unused function parameter.

```
function approveTransfer(address from, address to, uint256
    value, address spender) external pure override returns(
    bool)
function considerTransfer(address from, address to, uint256
    value) external pure override returns(bool)
```

**Recommendation**

Ignore warning in this case since the variable name has to be included in order to be able to override the original function implementation.

### 5.3.2 Mythril

No issues were detected.

### 5.3.3 Slither

No issues were detected.

## 5.4 ProxyAdmin.sol

### 5.4.1 Solidity Compiler

No issues were detected.

### 5.4.2 Mythril

No issues were detected.

### 5.4.3 Slither

No issues were detected.

wwwen.uni.lu/snt/research/sedan

JFK Building - 29, avenue J. F. Kennedy, L-1855 Luxembourg