# Introduction

Trufy was contracted by IVIRSE (Customer) to conduct a Smart Contract Code Review and Security Analysis. This report presents the findings of the security assessment of the Customer’s smart contract and its code review conducted between September 7th, 2022 - September 14th, 2022.

## Project Summary

* Project Name: IVIRSE
* Language: Solidity
* Codebase: <https://github.com/IVIRSE/IVIRSE>
* Commit: fc22999e16afe7ae1581f95f8c9b358bc0dcc773
* Audit method: Static Analysis, Manual Review
* Scope:
  + contracts/community/CampaignManagement.sol

## Vulnerability Summary

| Severity | # of Findings |
| --- | --- |
|  | 0 |
|  | 1 |
|  | 2 |
|  | 11 |

# Findings

| **ID** | **Title** | **Type** | **Severity** | **Resolved** |
| --- | --- | --- | --- | --- |
| [ID-01](#X9acfb7f0180bb159d065e75e27e3ebf66778278) | Project structure should be refined | Coding Style |  |  |
| [ID-02](#X9a77350eb9efdabd459a96af5e875261f675eba) | Lack of contract information in the error message of require() | Coding Style |  |  |
| [ID-03](#id-03-bad-modifier-naming) | Bad modifier naming | Coding Style |  |  |
| [ID-04](#X48067e6fec9651b60f3b1112ab8c8f6fe1dbfb5) | Bad gas optimization in function \_validateTimesAndAmounts() | Gas Optimization |  |  |
| [ID-05](#X24538f93e4e4e6ba5d1e6a2d6056449a6b21ce1) | Bad gas optimization in function \_validateAccountsAndAmounts() | Gas Optimization |  |  |
| [ID-06](#Xc081b7d379da19cf8a8f294194432749858abf3) | Inconsistency unit of parameter amount | Inconsistency |  |  |
| [ID-07](#id-07-bad-naming-convention) | Bad naming convention | Coding Style |  |  |
| [ID-08](#id-08-bad-function-naming) | Bad function naming | Coding Style |  |  |
| [ID-09](#X3e0a8c9cf62bdd7b4fa3224422305f18359d5c3) | Bad gas optimization in function \_getUsedTokenByStatus() and \_getAllowanceByStatus() | Gas Optimization |  |  |
| [ID-10](#X84f8a86160b4a35d876db79733daae68281eaa3) | Logic issue of function \_createCampaign() | Logical Issue |  |  |
| [ID-11](#id-11-bad-modifier-naming) | Bad modifier naming | Coding Style |  |  |
| [ID-12](#X2a39ec35a2bf330d380d817d01cfe366e0f7ecc) | Function \_getAllowanceByStatus() calculate wrong activeToken | Logical Issue |  |  |
| [ID-13](#Xc86e29baeeee92c30aa6b4e32f30fee4ca797e4) | Malicious admin can spam function createCampaign() | Logical Issue |  |  |
| [ID-14](#Xec609738c237f67d9f740cd7e37144ba8b6b98e) | Proper usage of public and external access | Gas optimization |  |  |

# Detailed Results

## ID-01: Project structure should be refined

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Coding Style | Informational |  |

### Description

All interfaces should be located in one directory.

### Recommendation

Recommended directory tree:

## ID-02: Lack of contract information in the error message of require()

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Coding Style | Informational | Lines 67, 81, 87, etc. |

### Description

In the error message, it is better to add more information about the contract, where the error is thrown.

Example in line 67:

### Recommendation

Adding contract name to the error message:

## ID-03: Bad modifier naming

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Coding Style | Informational | Line 102 |

### Description

Modifier isExist() can be bypassed if campaignName has not existed yet and vice versa.

### Recommendation

It is better to change name of modifier isExist() to ifNotExist().

## ID-04: Bad gas optimization in function \_validateTimesAndAmounts()

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Gas Optimization | Informational | Line 256 |

### Description

The function \_validateTimesAndAmounts() should require numberOfTime == numberOfAmount first, then require numberOfTime > 0 instead of require numberOfTime > 0 && numberOfAmount > 0

### Recommendation

## ID-05: Bad gas optimization in function \_validateAccountsAndAmounts()

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Gas Optimization | Informational | Line 278 |

### Description

The function \_validateAccountsAndAmounts() should require numberOfAccount == numberOfAmount first, then require numberOfAccount > 0, instead of require numberOfAccount > 0 && numberOfAmount > 0

### Recommendation

## ID-06: Inconsistency unit of parameter amount

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Inconsistency | Informational | Lines 113,258,280,297 |

### Description

Parameter amount represents the amount of tokens. In the function input, these parameters are not in the same unit.

For example, in the constructor, parameter amounts\_ is not multiplied by decimals . But in function \_createCampaign(), parameter \_amounts is in the form of multiplied by decimals .

This inconsistency may lead to confusion in programming.

### Recommendation

Parameter amount should be in the form of multiplied by decimals, in order to be able to represent the amount of tokens as a floating point number.

## ID-07: Bad naming convention

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Coding Style | Informational |  |

### Description

In constructor:

Parameter times\_ and amounts\_ have suffix \_.

In function \_createCampaign():

Parameter \_campaignName has prefix \_, but parameter releaseTime has no prefix.

### Recommendation

Name of parameters passed into functions should be unified one of the two conventions:

* Neither add underscore character \_ to prefix or suffix of parameters.
* Either add underscore character \_ to prefix or suffix of parameters.

## ID-08: Bad function naming

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Coding Style | Informational | Lines 230, 391 |

### Description

Name of function getTokenCanUse() in line 236 should be changed to getUseableTokenAmount.

Name of function \_getTokenMustNotUsed() in line 405 should be changed to  
\_getUnuseableTokenAmount.

## ID-09: Bad gas optimization in function \_getUsedTokenByStatus() and \_getAllowanceByStatus()

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Gas Optimization | Informational | Lines 403, 423 |

### Description

The declaration of parameters participants and participantLength may be redundant when if condition is not passed.

### Recommendation

The declaration of parameters participants and participantLength should be in the scope of if to optimize gas. These functions can be rewritten as follows:

## ID-10: Logic issue of function \_createCampaign()

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Logical Issue | Low | Line 294 |

### Description

In line 302:

Value campaignName is pushed into array \_campaignNames. Therefore, function \_getTokenCanUse() will iterate over this campaignName uselessly.

### Recommendation

Despite campaignName doesn’t have data in mapping \_campaigns, this is not optimal for gas and may lead to some logical issues in the future. It is better to move line 302

right below line 309

## ID-11: Bad modifier naming

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Coding Style | Informational | Line 87 |

### Description

AdminConsentStatus has three values: *NoAction*, *Reject*, *Accept*. It is more logical to change the name of modifer confirmedRelease to notRejectedRelease. It is compatible with modifer notConfirmedRelease.

## ID-12: Function \_getAllowanceByStatus() calculate wrong activeToken

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Logical Issue | Critical | Line 423 |

### Description

Function \_getAllowanceByStatus(status) calculates the total allowance amount of all participants in all campaign that has CampaignStatus = status.

In function \_getAllowanceByStatus(status) (line 423), \_token.allowance() will return token amount which an *account* can spend over all campaigns. By the following code

if an *account* joins more than one campaign, his activeToken will be plus more than one time of \_token.allowance().

Thus, function \_getAllowanceByStatus(status) is not worked as expected.

## ID-13: Malicious admin can spam function createCampaign()

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Logical Issue | Low | Line 129 |

### Description

Any malicious admin can spam function createCampaign() to create a campaign and decrease the amount of tokenCanUse. But in order to *Delete* these campaigns, need at least half of the admins vote *Reject* to *Delete* spam campaign and restore tokenCanUse.

## ID-14: Proper usage of public and external access

| **Type** | **Severity** | **Location** |
| --- | --- | --- |
| Gas Optimization | Informational |  |

### Description

The public functions:

* createCampaign() in line 129
* adminAcceptRelease() in line 141
* adminRejectRelease() in line 153
* release() in line 165
* deleteCampaign() in line 195
* getDatas() in line 209
* getCampaigns() in line 213
* getCampaign() in line 217

are only called from external, it is better to change from public to external for gas optimization.

# Appendix

**Severity Definitions**

| **Risk Level** | ### Description |
| --- | --- |
|  | This level vulnerabilities could be exploited easily and can lead to asset loss, data loss, asset, or data manipulation. They should be fixed right away. |
|  | This level vulnerabilities are hard to exploit but very important to fix, they carry an elevated risk of smart contract manipulation, which can lead to critical-risk severity. |
|  | This level vulnerabilities should be fixed, as they carry an inherent risk of future exploits, and hacks which may or may not impact the smart contract execution. |
|  | This level vulnerabilities can be ignored. They are code style violations and informational statements in the code. They may not affect the smart contract execution. |