Cyberscope

Audit Report

Fox&Gary

November 2023

Network	ETH
Address	0x9697f0899Ed3f1EF8266f1e09bC50496B2654F0b
Audited by	© cyberscope

Analysis

Critical Medium Minor / Informative Pass

Severity	Code	Description	Status
●	ST	Stops Transactions	Passed
●	OTUT	Transfers User's Tokens	Passed
●	ELFM	Exceeds Fees Limit	Passed
●	MT	Mints Tokens	Passed
●	BT	Burns Tokens	Passed
●	BC	Blacklists Addresses	Passed

Diagnostics

Critical Medium Minor / Informative

Severity	Code	Description	Status
Critical	IAA	Incorrect Amount Approval	Unresolved
Minor / Informative	MEE	Missing Events Emission	Unresolved
Minor / Informative	L04	Conformance to Solidity Naming Conventions	Unresolved
Minor / Informative	L07	Missing Events Arithmetic	Unresolved
Minor / Informative	L16	Validate Variable Setters	Unresolved

Table of Contents

Analysis 1
Diagnostics 2
Table of Contents 3
Review 4
Audit Updates 4
Source Files 5
Findings Breakdown 6
IAA - Incorrect Amount Approval 7
Description 7
Recommendation 7
MEE - Missing Events Emission 8
Description 8
Recommendation 8
L04 - Conformance to Solidity Naming Conventions 9
Description 9
Recommendation 10
L07 - Missing Events Arithmetic 11
Description 11
Recommendation 11
L16 - Validate Variable Setters 12
Description 12
Recommendation 12
Functions Analysis 13
Inheritance Graph 14
Flow Graph 15
Summary 16
Disclaimer 17
About Cyberscope 18

Review

Contract Name	FoxAndGary
Compiler Version	v0.8.19+commit.7dd6d404
Optimization	200 runs
Explorer	https://etherscan.io/address/0x9697f0899ed3f1ef8266f1e09bc50496b2654f0b
Address	0x9697f0899ed3f1ef8266f1e09bc50496b2654f0b
Network	ETH
Symbol	FNG
Decimals	18
Total Supply	10,000,000,000,000

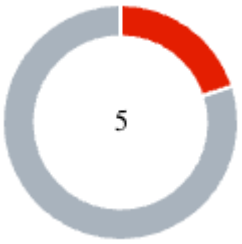
Audit Updates

Initial Audit	06 Nov 2023

Source Files

Filename	SHA256
contracts/Owned.sol	e3a1c495ddb377a5bb958c16a6ba2f2b76d31be944ac03d43afe18d2411cd283
contracts/FoxAndGary.sol	24cdab72ffe0b08e8f5fb9cb1db68516dcd00fa702c6370b4e22c0c147343429
contracts/ERC20.sol	0d7993caf9532a99790bd9c94bff244ea68ab4f5b0123bb9510653474b063edb

Findings Breakdown



Critical	1
Medium	0
Minor / Informative	4

Severity	Unresolved	Acknowledged	Resolved	Other
Critical	1	0	0	0
Medium	0	0	0	0
Minor / Informative	4	0	0	0

IAA - Incorrect Amount Approval

Criticality	Critical
Location	contracts/FoxAndGary.sol#L29
Status	Unresolved

Description

The contract exhibits a critical vulnerability in the `transferFrom` function, which is a fundamental part of the ERC20 token transfer flow. In this implementation, the provided `_amount` is processed through the `_processFee` function, which deducts the transaction fee from the amount. This action leads to the `transferFrom` method receiving an amount that is less than the actual intended transaction value. Consequently, when a token owner approves a spender to transfer a certain amount of tokens, the approved amount will be incorrect due to the deduction of fees.

This vulnerability can have severe consequences, including potential misuse or unintended transactions due to the discrepancy between the approved amount and the actual amount deducted. Such issues may lead to confusion and financial losses for token holders.

```
function transferFrom(address _from, address _to, uint256 _amount) public
override returns (bool) {
    return ERC20.transferFrom(
        _from,
        _to,
        _processFee(_from, _amount)
    );
}
```

Recommendation

The team is advised to take these segments into consideration and make adjustments to the `transferFrom` function. The adjustments should make sure the token owner approves the correct amount for the spender. By implementing such changes, the team can rectify the vulnerability and provide token holders with a secure and accurate means of approving transactions. This will prevent potential financial losses and misuses stemming from the incorrect approval of token amounts.

MEE - Missing Events Emission

Criticality	Minor / Informative
Location	contracts/FoxAndGary.sol#L53,38
Status	Unresolved

Description

The contract performs actions and state mutations from external methods that do not result in the emission of events. Emitting events for significant actions is important as it allows external parties, such as wallets or dApps, to track and monitor the activity on the contract. Without these events, it may be difficult for external parties to accurately determine the current state of the contract.

```
treasury = _treasury;
ERC20.balanceOf[treasury] += feeAmount;
```

Recommendation

It is recommended to include events in the code that are triggered each time a significant action is taking place within the contract. These events should include relevant details such as the user's address and the nature of the action taken. By doing so, the contract will be more transparent and easily auditable by external parties. It will also help prevent potential issues or disputes that may arise in the future.

L04 - Conformance to Solidity Naming Conventions

Criticality	Minor / Informative
Location	contracts/FoxAndGary.sol#L18,25,43,47,52
Status	Unresolved

Description

The Solidity style guide is a set of guidelines for writing clean and consistent Solidity code. Adhering to a style guide can help improve the readability and maintainability of the Solidity code, making it easier for others to understand and work with.

The followings are a few key points from the Solidity style guide:

1. Use camelCase for function and variable names, with the first letter in lowercase (e.g., myVariable, updateCounter).
2. Use PascalCase for contract, struct, and enum names, with the first letter in uppercase (e.g., MyContract, UserStruct, ErrorEnum).
3. Use uppercase for constant variables and enums (e.g., MAX_VALUE, ERROR_CODE).
4. Use indentation to improve readability and structure.
5. Use spaces between operators and after commas.
6. Use comments to explain the purpose and behavior of the code.
7. Keep lines short (around 120 characters) to improve readability.

```
address _to
uint256 _amount
address _from
uint256 _fee
address _treasury
```

Recommendation

By following the Solidity naming convention guidelines, the codebase increased the readability, maintainability, and makes it easier to work with.
Find more information on the Solidity documentation
https://docs.soliditylang.org/en/v0.8.17/style-guide.html#naming-convention.

L07 - Missing Events Arithmetic

Criticality	Minor / Informative
Location	contracts/FoxAndGary.sol#L49
Status	Unresolved

Description

Events are a way to record and log information about changes or actions that occur within a contract. They are often used to notify external parties or clients about events that have occurred within the contract, such as the transfer of tokens or the completion of a task.

It's important to carefully design and implement the events in a contract, and to ensure that all required events are included. It's also a good idea to test the contract to ensure that all events are being properly triggered and logged.

```
fee = _fee
```

Recommendation

By including all required events in the contract and thoroughly testing the contract's functionality, the contract ensures that it performs as intended and does not have any missing events that could cause issues with its arithmetic.

L16 - Validate Variable Setters

Criticality	Minor / Informative
Location	contracts/FoxAndGary.sol#L53
Status	Unresolved

Description

The contract performs operations on variables that have been configured on user-supplied input. These variables are missing of proper check for the case where a value is zero. This can lead to problems when the contract is executed, as certain actions may not be properly handled when the value is zero.

```
treasury = _treasury
```

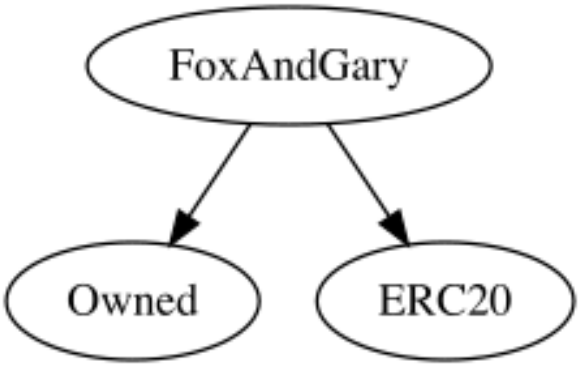
Recommendation

By adding the proper check, the contract will not allow the variables to be configured with zero value. This will ensure that the contract can handle all possible input values and avoid unexpected behavior or errors. Hence, it can help to prevent the contract from being exploited or operating unexpectedly.

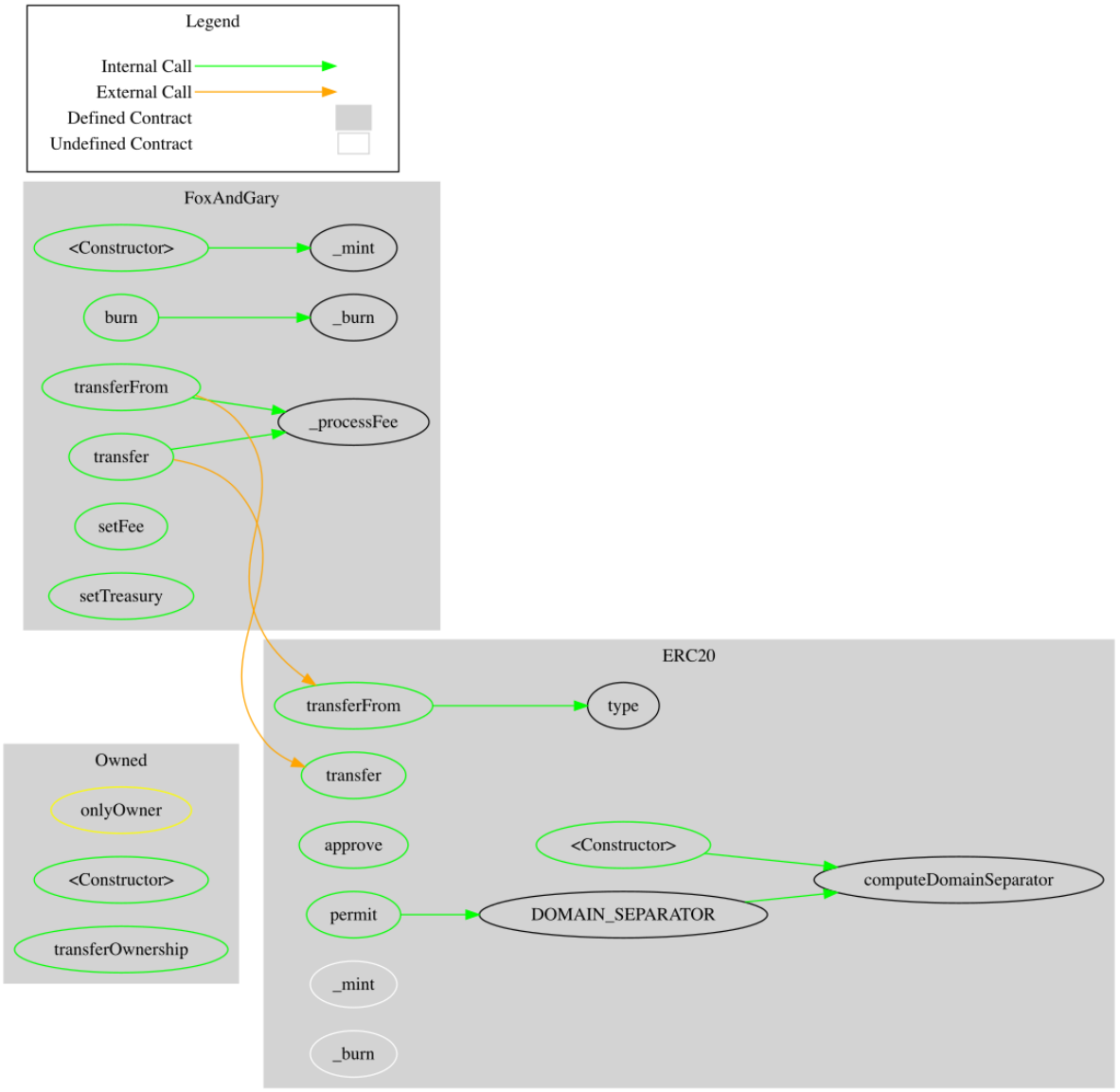
Functions Analysis

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
FoxAndGary	Implementation	ERC20, Owned		
		Public	✓	ERC20 Owned
	transfer	Public	✓	-
	transferFrom	Public	✓	-
	_processFee	Private	✓	
	burn	Public	✓	-
	setFee	Public	✓	onlyOwner
	setTreasury	Public	✓	onlyOwner

Inheritance Graph



Flow Graph


Legend
Internal Call
External Call
Defined Contract
Undefined Contract
FoxAndGary
<Constructor>
_mint
burn
_burn
transferFrom
_processFee
transfer
setFee
setTreasury
ERC20
transferFrom
type
transfer
approve
<Constructor>
computeDomainSeparator
permit
DOMAIN_SEPARATOR
_mint
_burn
Owned
onlyOwner
<Constructor>
transferOwnership

Summary

Fox&Gary contract implements a token mechanism. This audit investigates security issues, business logic concerns, and potential improvements. Fox&Gary is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler errors and one critical issue. The contract Owner can access some admin functions that can not be used in a malicious way to disturb the users' transactions. There is also a limit of max 2% fees.

Disclaimer

The information provided in this report does not constitute investment, financial or trading advice and you should not treat any of the document's content as such. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes nor may copies be delivered to any other person other than the Company without Cyberscope's prior written consent. This report is not nor should be considered an "endorsement" or "disapproval" of any particular project or team. This report is not nor should be regarded as an indication of the economics or value of any "product" or "asset" created by any team or project that contracts Cyberscope to perform a security assessment. This document does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors' business, business model or legal compliance. This report should not be used in any way to make decisions around investment or involvement with any particular project. This report represents an extensive assessment process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk Cyberscope's position is that each company and individual are responsible for their own due diligence and continuous security Cyberscope's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies and in no way claims any guarantee of security or functionality of the technology we agree to analyze. The assessment services provided by Cyberscope are subject to dependencies and are under continuing development. You agree that your access and/or use including but not limited to any services reports and materials will be at your sole risk on an as-is where-is and as-available basis Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives false negatives and other unpredictable results. The services may access and depend upon multiple layers of third parties.

About Cyberscope

Cyberscope is a blockchain cybersecurity company that was founded with the vision to make web3.0 a safer place for investors and developers. Since its launch, it has worked with thousands of projects and is estimated to have secured tens of millions of investors' funds.

Cyberscope is one of the leading smart contract audit firms in the crypto space and has built a high-profile network of clients and partners.

The Cyberscope team

https://www.cyberscope.io