Cyberscope

Audit Report

Xocolatl Assets Accountant

October 2022

Github	https://github.com/La-DAO/xocolatl-contracts
Commit	c367fec4a276bece4e580aca4a26e2147eb09643
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	AssetsAccountant
Compiler Version	v0.8.13+commit.abaa5c0e
Optimization	200 runs
Github	https://github.com/La-DAO/xocolatl-contracts/blob/main/contracts/AssetsAccountant.sol
Commit	c367fec4a276bece4e580aca4a26e2147eb09643
Explorer	https://testnet.bscscan.com/token/0x397bf582a7467a9901Ca25E88ff4e9bf42160C2d
Domain	https://xocolatl.club

Audit Updates

Initial Audit	26th October 2022
Corrected	

Source Files

Filename	SHA256
@openzeppelin/contracts/access/AccessControl.sol	5af1771388b4fe634e0a566716e32c6d00a5372875099127b274d4cf8a94e9d2
@openzeppelin/contracts/access/IAccessControl.sol	d03c1257f2094da6c86efa7aa09c1c07ebd33dd31046480c5097bc2542140e45
@openzeppelin/contracts/token/ERC1155/ERC1155.sol	3a7b1481259da24728a0bac33ac9728c0faf71d436e4f198209815f732240a24
@openzeppelin/contracts/token/ERC1155/extensions/IERC1155MetadataURI.sol	6987fbfa647d3da51e8c270371ac48c5fcd26fb046cf54644b39aa098ae30324
@openzeppelin/contracts/token/ERC1155/IERC1155.sol	fd6a1801f1f2f8af0a3ece0b254da06ec24568aec02cfe94827061379aebc6f3
@openzeppelin/contracts/token/ERC1155/IERC1155Receiver.sol	578834a1bcdac6a22de5e07ae63bbbd4d41615f35950afc6e6c068d92619b334
@openzeppelin/contracts/utils/Address.sol	1e0922f6c0bf6b1b8b4d480dcabb691b1359195a297bde6dc5172e79f3a1f826
@openzeppelin/contracts/utils/Context.sol	1458c260d010a08e4c20a4a517882259a23a4baa0b5bd9add9fb6d6a1549814a
@openzeppelin/contracts/utils/introspection/ERC165.sol	8806a632d7b656cadb8133ff8f2acae4405b3a64d8709d93b0fa6a216a8a6154
@openzeppelin/contracts/utils/introspection/IERC165.sol	701e025d13ec6be09ae892eb029cd83b3064325801d73654847a5fb11c58b1e5

@openzeppelin/contracts/utils/Strings.sol	34127ad0054df5963b0fd694c1b313d17e9114a2f426b85526d6d976210298ab
contracts/AssetsAccountant.sol	f04e37551d1868a9dbd65cc2c71accdd43b08da1c957a5dceeaed8b8b3f45822
contracts/interfaces/IHouseOfCoinState.sol	7f3f45d5b52459c1700f70df4a60871495500cfaceb048bce25404fadfa7f030
contracts/interfaces/IHouseOfReserve.sol	2cf3c1454c96809fe84a571802268e15539652ab80328dbc7cd99b1db5f7997e
contracts/interfaces/IOracle.sol	1f13347804c9d374a356eb2c5100a4f983c3873c164e5bd1d3890d79bc3786a4

Introduction

The AssetsAccountant contract implements the ERC1155 standard. It is responsible for keeping all the reserved and backed token ids.

House Registry

The contract tracks all the “house of reserve” and “house of coin” contracts. The contract owner is responsible for registering all the ‘house’ contracts.

Data

The contract keeps track of data by keeping four registries:

- houseOfReserves
- reservesIds
- houseOfCoins
- _isARegisteredHouse

Roles

The contract has 5 roles:

- DEFAULT_ADMIN_ROLE
- URI_SETTER_ROLE
- MINTER_ROLE
- BURNER_ROLE
- LIQUIDATOR_ROLE

Contract Diagnostics

● Critical ● Medium ● Minor / Informative

Severity	Code	Description	Status
●	TCP	Transfer Caller Permit	Unresolved
●	RFP	Redundant Function Parameter	Unresolved
●	MT	Mints Tokens	Unresolved
●	BT	Burns Tokens	Unresolved
●	L03	Redundant Statements	Unresolved
●	L04	Conformance to Solidity Naming Conventions	Unresolved

TCP - Transfer Caller Permit

Criticality	minor / informative
Location	contracts/AssetsAccountant.sol#L205
Status	Unresolved

Description

The LIQUIDATOR role has the authority to transfer tokens. The ‘safeTransferFrom’ method is solery called by the ‘HouseOfCoin’ addresses. As an extra validation step, the contract could check if the sender is registered in the ‘_isARegisteredHouse’ property.

```
function safeTransferFrom(
    address from,
    address to,
    uint256 id,
    uint256 amount,
    bytes calldata data
) public override onlyRole(LIQUIDATOR_ROLE) {
    _safeTransferFrom(from, to, id, amount, data);
}
```

Recommendation

The contract could remove the second parameter ‘asset’ and take the address value from the ‘houseAddress’ parameter.

RFP - Redundant Function Parameter

Criticality	minor / informative
Location	contracts/AssetsAccountant.sol#L70
Status	Unresolved

Description

The admin role has the ability to register a ‘house’. The method accepts two arguments. The ‘house’ and the ‘asset’ address. The ‘asset’ address is the ‘reserved’ and the ‘backed’ addresses from the corresponding house. Each house holds its asset address. This address is the same as the second parameter ‘asset’.

```
function registerHouse(address houseAddress, address asset)
    external
    onlyRole(DEFAULT_ADMIN_ROLE)
{
   ...
}
```

Recommendation

The contract could remove the second parameter ‘asset’ and take the address value from the ‘houseAddress’ parameter.

MT - Mints Tokens

Criticality	minor / informative
Location	contract.sol#L158,168
Status	Unresolved

Description

The MINTER role has the authority to mint tokens. The owner may take advantage of it by calling the `mint` or the `mintBatch` functions.

```
function mint(address account, uint256 id, uint256 amount, bytes memory data)
    external
    onlyRole(MINTER_ROLE)
{
    _mint(account, id, amount, data);
}
function mintBatch(address to, uint256[] memory ids, uint256[] memory amounts,
bytes memory data)
    external
    onlyRole(MINTER_ROLE)
{
    _mintBatch(to, ids, amounts, data);
}
```

Recommendation

The owner should carefully manage the credentials of the owner's account. We advised considering an extra-strong security mechanism that the actions may be quarantined by many users instead of one. The owner could also renounce the contract ownership for a period of time or pass the access to the zero address.

BT - Burns Tokens

Criticality	minor / informative
Location	contract.sol#L182,193
Status	Unresolved

Description

The BURNER role has the authority to burn tokens from a specific address. The owner may take advantage of it by calling the `burn` or the `burnBatch` functions.

```
function burn(
    address account,
    uint256 id,
    uint256 amount
) public onlyRole(BURNER_ROLE) {
    _burn(account, id, amount);
}

function burnBatch(
    address account,
    uint256[] memory ids,
    uint256[] memory amounts
) public onlyRole(BURNER_ROLE) {
    _burnBatch(account, ids, amounts);
}
```

Recommendation

The owner should carefully manage the credentials of the owner's account. We advised considering an extra-strong security mechanism that the actions may be quarantined by many users instead of one. The owner could also renounce the contract ownership for a period of time or pass the access to the zero address.

L03 - Redundant Statements

Criticality	minor / informative
Location	contracts/AssetsAccountant.sol#L226,228,227,229,225
Status	Unresolved

Description

The contract contains statements that are not used and have no effect. As a result, those segments increase the code size of the contract unnecessarily.

```
from;
to;
ids;
amounts;
data;
```

Recommendation

Remove the redundant statements in order to decrease the code size.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor / informative
Location	contracts/AssetsAccountant.sol#L29
Status	Unresolved

Description

Solidity defines a naming convention that should be followed. Rule exceptions:

- Allow constant variable name/symbol/decimals to be lowercase.
- Allow _ at the beginning of the mixed_case match for private variables and unused parameters.

```
_isARegisteredHouse
```

Recommendation

Follow the Solidity naming convention.
https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions.

Contract Functions

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
AccessControl	Implementation	Context, IAccessControl, ERC165		
	supportsInterface	Public		-
	hasRole	Public		-
	_checkRole	Internal		
	_checkRole	Internal		
	getRoleAdmin	Public		-
	grantRole	Public	✓	onlyRole
	revokeRole	Public	✓	onlyRole
	renounceRole	Public	✓	-
	_setupRole	Internal	✓	
	_setRoleAdmin	Internal	✓	
	_grantRole	Internal	✓	
	_revokeRole	Internal	✓	
IAccessControl	Interface			
	hasRole	External		-
	getRoleAdmin	External		-
	grantRole	External	✓	-
	revokeRole	External	✓	-
	renounceRole	External	✓	-
ERC1155	Implementation	Context, ERC165, IERC1155, IERC1155MetadataURI		
	<Constructor>	Public	✓	-

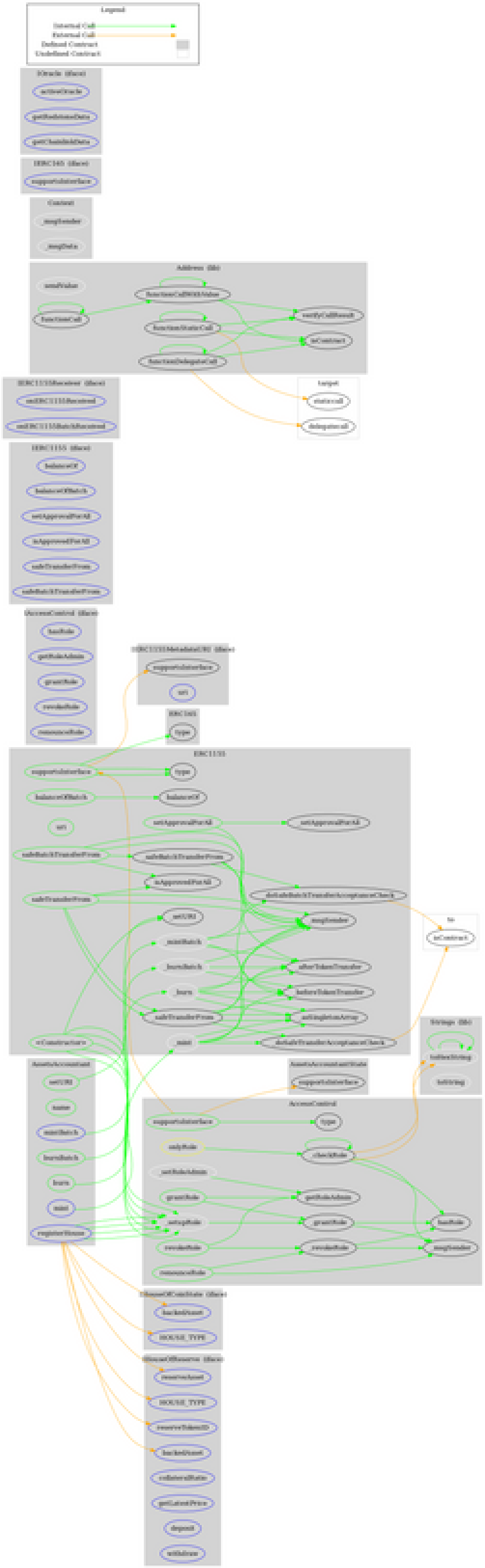
	supportsInterface	Public		-
	uri	Public		-
	balanceOf	Public		-
	balanceOfBatch	Public		-
	setApprovalForAll	Public	✓	-
	isApprovedForAll	Public		-
	safeTransferFrom	Public	✓	-
	safeBatchTransferFrom	Public	✓	-
	_safeTransferFrom	Internal	✓	
	_safeBatchTransferFrom	Internal	✓	
	_setURI	Internal	✓	
	_mint	Internal	✓	
	_mintBatch	Internal	✓	
	_burn	Internal	✓	
	_burnBatch	Internal	✓	
	_setApprovalForAll	Internal	✓	
	_beforeTokenTransfer	Internal	✓	
	_afterTokenTransfer	Internal	✓	
	_doSafeTransferAcceptanceCheck	Private	✓	
	_doSafeBatchTransferAcceptanceCheck	Private	✓	
	_asSingletonArray	Private		
IERC1155MetadataURI	Interface	IERC1155		
	uri	External		-
IERC1155	Interface	IERC165		
	balanceOf	External		-
	balanceOfBatch	External		-
	setApprovalForAll	External	✓	-
	isApprovedForAll	External		-
	safeTransferFrom	External	✓	-
	safeBatchTransferFrom	External	✓	-

IERC1155Receiver	Interface	IERC165		
	onERC1155Received	External	✓	-
	onERC1155BatchReceived	External	✓	-
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	verifyCallResult	Internal		
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
ERC165	Implementation	IERC165		
	supportsInterface	Public		-
IERC165	Interface			
	supportsInterface	External		-
Strings	Library			
	toString	Internal		
	toHexString	Internal		
	toHexString	Internal		
	toHexString	Internal		
AssetsAccountantState	Implementation			

AssetsAccountant	Implementation	ERC1155, AccessControl, AssetsAccountantState		
	<Constructor>	Public	✓	ERC1155
	registerHouse	External	✓	onlyRole
	name	Public		-
	setURI	Public	✓	onlyRole
	mint	External	✓	onlyRole
	mintBatch	External	✓	onlyRole
	burn	Public	✓	onlyRole
	burnBatch	Public	✓	onlyRole
	safeTransferFrom	Public	✓	onlyRole
	safeBatchTransferFrom	Public		-
	supportsInterface	Public		-
IHouseOfCoinState	Interface			
	HOUSE_TYPE	External	✓	-
	backedAsset	External		-
IHouseOfReserve	Interface	IOracle		
	reserveAsset	External		-
	backedAsset	External		-
	reserveTokenID	External		-
	HOUSE_TYPE	External	✓	-
	collateralRatio	External		-
	getLatestPrice	External		-
	deposit	External	✓	-
	withdraw	External	✓	-
IOracle	Interface			
	activeOracle	External		-
	getRedstoneData	External		-

	getChainlinkData	External		-

Contract Flow



Domain Info

Domain Name	xocolatl.club
Registry Domain ID	D017C2E7D305043B48BB9BAC3CE267A07-GDREG
Creation Date	2022-09-09T07:58:44Z
Updated Date	2022-09-14T07:58:44Z
Registry Expiry Date	2023-09-09T07:58:44Z
Registrar WHOIS Server	whois.opensrs.net
Registrar URL	www.opensrs.com
Registrar	Tucows Domains Inc.
Registrar IANA ID	69

The domain was created about 1 month before the creation of the audit. It will expire in 11 months.

There is no public billing information, the creator is protected by the privacy settings.

Summary

The HouseOfCoin contract implements a multi-token standard. It operates as the accountant of the Xocolatl Ecosystem. This audit investigates security issues and mentions business logic concerns and potential improvements.

Disclaimer

All the content provided in this document is for general information only and should not be used as financial advice or a reason to buy any investment.

Cyberscope team provides no guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Always Do your own research and protect yourselves from being scammed.

The Cyberscope team has audited this project for general information and only expresses their opinion based on similar projects and checks from popular diagnostic tools. Under no circumstances did Cyberscope receive a payment to manipulate those results or change the awarding badge that we will be adding to our website.

Always Do your own research and protect yourselves from scams. This document should not be presented as a reason to buy or not buy any particular token.

The Cyberscope team disclaims any liability for the resulting losses.

About Cyberscope

Coinscope audit and K.Y.C. service has been rebranded to Cyberscope.

Coinscope is the leading early coin listing, voting and auditing authority firm. The audit process is analyzing and monitoring many aspects of the project. That way, it gives the community a good sense of security using an informative report and a generic score.

Cyberscope and Coinscope are aiming to make crypto discoverable and efficient globally. They provide all the essential tools to assist users draw their own conclusions.

The Cyberscope team

https://www.cyberscope.io