Cyberscope

Audit Report

Xocolatl Assets Accountant

December 2022

Github	https://github.com/La-DAO/xocolatl-contracts
Commit	7d780e9a7573b88f042f8f45096a201442ea782e
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	AssetsAccountant
Testing Deploy	https://testnet.bscscan.com/address/0xac34eeb79854d7f190f7d9452e058e5c5581ef01

Audit Updates

Initial Audit	26 Oct 2022 https://github.com/cyberscope-io/audits/blob/main/xocolatl/v1/assetsAccountant.pdf
Corrected Phase 2	19 Dec 2022

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	122eae76d48042142db0469395fa6721d60b5105ee457e36f8e4a8b43fb98b23

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
	MT	Mints Tokens	Unresolved
	BT	Burns Tokens	Unresolved
	L18	Multiple Pragma Directives	Unresolved

MT - Mints Tokens

Criticality	Minor / Informative
Location	contract.sol#L164,176
Status	Unresolved

Description

The MINTER role has the authority to burn tokens from a specific address. 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 team should carefully manage the private keys of the owner’s account. We strongly recommend a powerful security mechanism that will prevent a single user from accessing the contract admin functions. That risk can be prevented by temporarily locking the contract or renouncing ownership.

BT - Burns Tokens

Criticality	Minor / Informative
Location	contract.sol#L192,203
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 team should carefully manage the private keys of the owner’s account. We strongly recommend a powerful security mechanism that will prevent a single user

from accessing the contract admin functions. That risk can be prevented by temporarily locking the contract or renouncing ownership.

L18 - Multiple Pragma Directives

Criticality	Minor / Informative
Location	contracts/AssetsAccountant.sol#L2
Status	Unresolved

Description

If the contract includes multiple conflicting pragma directives, it may produce unexpected errors. To avoid this, it's important to include the correct pragma directive at the top of the contract and to ensure that it is the only pragma directive included in the contract.

```
pragma solidity 0.8.13;
```

Recommendation

It is important to include only one pragma directive at the top of the contract and to ensure that it accurately reflects the version of Solidity that the contract is written in. By including all required compiler options and flags in a single pragma directive, you can avoid conflicts and ensure that the contract can be compiled correctly.

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		
		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	✓	-

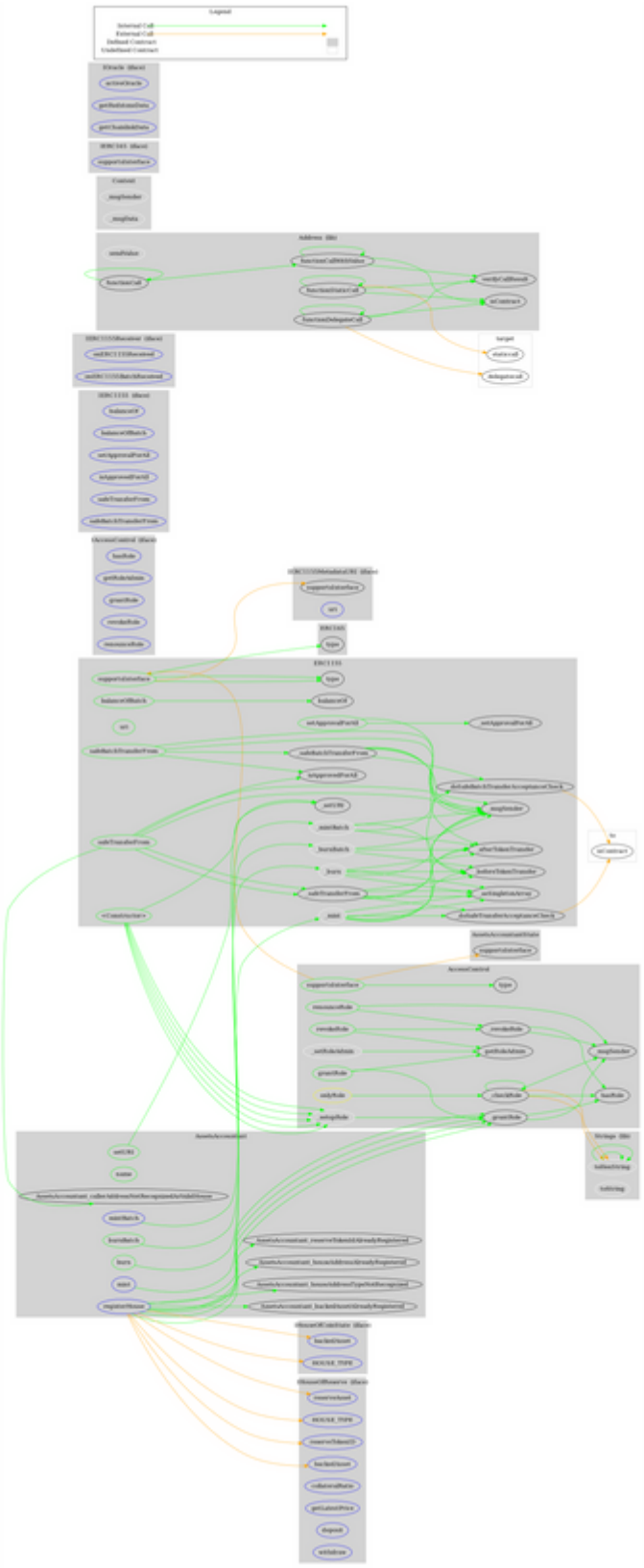
	safeBatchTransferFrom	External	✓	-
IERC1155Receiver	Interface	IERC165		
	onERC1155Received	External	✓	-
	onERC1155BatchReceived	External	✓	-
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	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		
		Public	✓	ERC1155

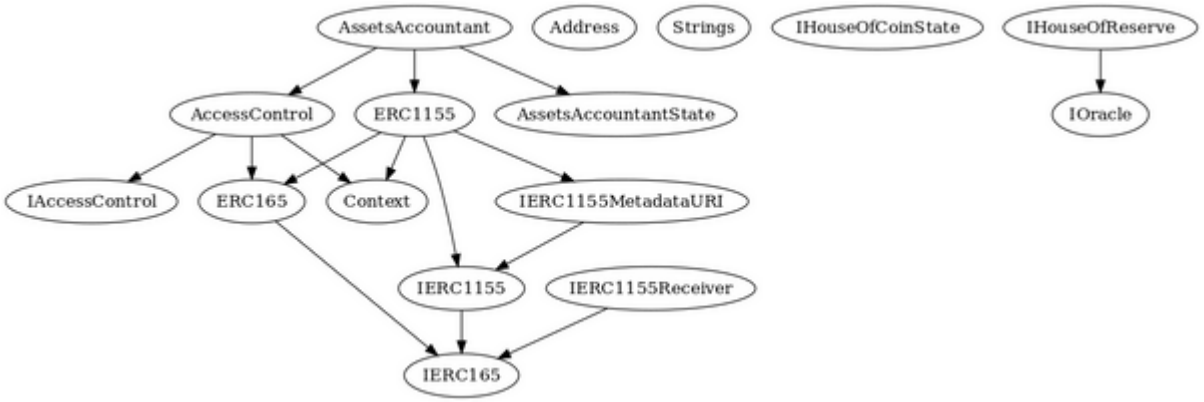
	registerHouse	External	✓	onlyRole
	name	Public		-
	setURI	Public	✓	onlyRole
	mint	External	✓	onlyRole
	mintBatch	External	✓	onlyRole
	burn	Public	✓	onlyRole
	burnBatch	Public	✓	onlyRole
	safeTransferFrom	Public	✓	onlyRole
	safeBatchTransferFrom	Public		-
	supportsInterface	Public		-
IHouseOfCoinState	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



Inheritance Graph



Summary

The AssetsAccountant 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

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