
DLYCOP TOKEN SMART CONTRACT AUDIT

August 2021

BLOCKCHAIN CONSILIUM



Contents

Disclaimer	3
Purpose of the report	3
Introduction	4
Audit Summary	4
Overview	5
Methodology	5
Classification / Issue Types Definition	5
Smart Contract Overview.....	6
Attacks & Issues considered while auditing	7
Overflows and underflows	7
Reentrancy Attack	7
Replay attack	7
Short address attack	8
Approval Double-spend	9
Sybil attacks	10
Issues Found	11
High Severity Issues	11
Moderate Severity Issues	11
Low Severity Issues	11
Informational Observations.....	11
Improvement Suggestions	13
Ability to Reclaim Tokens transferred to Token Smart contract	13
Appendix.....	14
Smart Contract Summary	14
Relayer	14
DailyCopTokenRoot.....	29
DailyCopTokenChild	36
Inheritance Graph and UML Diagram.....	57
Slither Results	58
Relayer	58
DailyCopTokenRoot.....	74
DailyCopTokenChild	78



Disclaimer

THE AUDIT MAKES NO STATEMENTS OR WARRANTIES ABOUT UTILITY OF THE CODE, SAFETY OF THE CODE, SUITABILITY OF THE BUSINESS MODEL, REGULATORY REGIME FOR THE BUSINESS MODEL, OR ANY OTHER STATEMENTS ABOUT FITNESS OF THE CONTRACTS TO PURPOSE, OR THEIR BUG FREE STATUS. THE AUDIT DOCUMENTATION IS FOR DISCUSSION PURPOSES ONLY.

THE CONTENT OF THIS AUDIT REPORT IS PROVIDED "AS IS", WITHOUT REPRESENTATIONS AND WARRANTIES OF ANY KIND, AND BLOCKCHAIN CONSILIUM DISCLAIMS ANY LIABILITY FOR DAMAGE ARISING OUT OF, OR IN CONNECTION WITH, THIS AUDIT REPORT. COPYRIGHT OF THIS REPORT REMAINS WITH BLOCKCHAIN CONSILIUM.

Purpose of the report

The Audits and the analysis described therein are created solely for Clients and published with their consent. The scope of our review is limited to a review of Solidity code and only the Solidity code we note as being within the scope of our review within this report. The Solidity language itself remains under development and is subject to unknown risks and flaws. The review does not extend to the compiler layer, or any other areas beyond the Solidity programming language that could present security risks. Cryptographic tokens and smart contracts are emergent technologies and carry with them high levels of technical risk and uncertainty.

The Audits are not an endorsement or indictment of any particular project or team, and the Audits do not guarantee the security of any particular project. This Report does not consider, and should not be interpreted as considering or having any bearing on, the potential economics of a token, token sale or any other product, service or other asset. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. No Report provides any warranty or representation to any Third-Party in any respect, including regarding the bugfree nature of code, the business model or proprietors of any such business model, and the legal compliance of any such business. No third party should rely on the Audits in any way, including for the purpose of making any decisions to buy or sell any token, product, service or other asset. This Report does not constitute investment advice, is not intended to be relied upon as investment advice, is not an endorsement of this project or team, and it is not a guarantee as to the absolute security of the project. There is no owed duty to any Third-Party by virtue of publishing these Audits.



Introduction

We first thank b4vtechnologies.nl for giving us the opportunity to audit their smart contract. This document outlines our methodology, audit details, and results.

b4vtechnologies.nl asked us to review their DLYCOP token smart contract (GitHub commit hash: f57fb93c5d6842d51c1b931233489bea1d9b206d). [Blockchain Consilium](#) reviewed the system from a technical perspective looking for bugs, issues and vulnerabilities in their code base. The Audit is valid for the mentioned GitHub commit hash(es) only. The audit is not valid for any other versions of the smart contract. Read more below.

Audit Summary

This code is clean, thoughtfully written and in general well architected. The code conforms closely to the documentation and specification.

Overall, the code is clear on what it is supposed to do for each function. The visibility and state mutability of all the functions are clearly specified, and there are no confusions.

Audit Scope & Info	
Platform	Polygon & Ethereum
Language	Solidity
Audit Method	Whitebox
Repository	https://github.com/Mi-Daily/DLY-COP/tree/f57fb93c5d6842d51c1b931233489bea1d9b206d
SHA256	<pre>> shasum -a 256 IChildToken.sol IDailyToken.sol IMintableERC20.sol Relay.sol DailyCopTokenChild.sol DailyCopTokenRoot.sol c9e5d6fe0b4d9469bd3182f8c114bcb634cb3cf5254682750edbb1c0a9c4ebee 42be09217672da429d42b2ee4817d2d0082cf01332e0b32dc8cabb5b216a3c03 8fbbbb72d2b86eadfc1842632609895f1750d54f4e63928c0bd64d3f1fc963d9 a2cfd72c293643fc03b1eae0e6a4ba7490f7bfb4fbc0f146bac2fba8ddaa4dc 722f571c0715eea1cd51a8c9bbe615acb7e4a74bb0f43d150f0a374b934b7120 66df266066275930c6d7fe16991f57a85d93b655a89fe9a6029e07430e519baf</pre>
Audit Results & Findings	
High Severity Issues	None
Moderate Severity Issues	None
Low Severity Issues	2
Improvement Suggestions	1
Informational Observations	4



Overview

The project has six Solidity files for the DLYCOP ERC20 Token Smart Contract Setup, `IChildToken.sol` `IDailyToken.sol` `IMintableERC20.sol` `Relayer.sol` `DailyCopTokenChild.sol` `DailyCopTokenRoot.sol`. We manually reviewed each line of code in the smart contracts.

The smart contract is an ERC20 token based on OpenZeppelin and implements additional functions for mitigating approval double-spend attacks. It also implements necessary functions for Polygon PoS bridge integration, snapshot feature, ERC20 Permit feature which enables the admin / external software to pay for gas on behalf of the token holder.

The relayer contract is upgradeable, and the token contracts are uncapped mintable, this introduces some level of centralization in the smart contracts.

The smart contract includes imports from OpenZeppelin libraries, code review of OpenZeppelin libraries or external contracts are outside the scope of this audit report.

Methodology

Blockchain Consilium manually reviewed the smart contract line-by-line, keeping in mind industry best practices and known attacks, looking for any potential issues and vulnerabilities, and areas where improvements are possible.

We also used automated tools like slither and surya for analysis and reviewing the smart contract. The raw output of these tools is included in the Appendix. These tools often give false-positives, and any issues reported by them but not included in the issue list can be considered not valid.

Classification / Issue Types Definition

1. **High Severity:** which presents a significant security vulnerability or failure of the contract across a range of scenarios, or which may result in loss of funds.
2. **Moderate Severity:** which affects the desired outcome of the contract execution or introduces a weakness that can be exploited. It may not result in loss of funds but breaks the functionality or produces unexpected behaviour.
3. **Low Severity:** which does not have a material impact on the contract execution and is likely to be subjective.

The smart contract is considered to pass the audit, as of the audit date, if no high severity or moderate severity issues are found.



Smart Contract Overview

The following overview was submitted by b4vtechnologies.nl

This repository contains the token and relayer contracts of the Daily COP (Colombian Peso) stablecoin.

The DLY COP token is an ERC20 token that:

- can be minted by an account with the MINTER_ROLE. Initially this will be centralized and done by a multisig address controlled by different stakeholders in Daily, in a later phase the token will be collateralized on-chain and minting will only be possible with a deposit of collateral, making it more decentralized.*
- can be burned by the same token holder or anyone who was approved the allowance of the token.*
- has the permit functionality as described in <https://eips.ethereum.org/EIPS/eip-2612> [EIP-2612] allowing approvals to be made via signatures so we can more easily create services to cover for the transaction costs of users.*
- has snapshot functionality that can be used in case any migration has to be done or any rewards need to be given to token holders.*
- will be deployed on the layer 2 solution Polygon with the contract "DailyCopTokenChild" and on layer 1 Ethereum with the contract "DailyCopTokenRoot".*
- can be moved to and from layer 1 (Ethereum) using the Matic Bridge, for which the deposit and withdrawal functions are implemented in the Child token contract and the mint function in the Root token contract is limited to the Predicate Proxy contract of the Bridge.*
- currently has a DEFAULT_ADMIN_ROLE to control and change the roles (MINTER_ROLE, SNAPSHOT_ROLE and DEPOSITOR_ROLE) in case it is necessary. Once the collateralization protocol is implemented and minting is done automatically through the protocol, the DEFAULT_ADMIN_ROLE may disappear by making the only granted account renounce from it.*

The Relayer contract is added so that transactions can be signed by users and relayed by other accounts (trusted relayers with the RELAYER_ROLE) that will pay for the gas fee. It makes use of the permit function in the token contract and makes it possible to do a permit and transferFrom in one and the same transaction. In the future we will charge a fee in the own DLY COP token, which will be transparent in the Relayer contract. To be able to update this fee model, this contract is upgradeable. In the future we will not only have transfers with permit, but any action that can be done with our token, for example exchanges, without the need for the user to have the native network



currency. The contract will also be improved so the relaying can be done by any account instead of needing the RELAYER_ROLE, but for now the RELAYER_ROLE will only be given to accounts of trusted services of Daily, so we can make sure the tokens are sent to the intended recipient, since this is not part of the permit signature.

Attacks & Issues considered while auditing

In order to check for the security of the contract, we reviewed each line of code in the smart contract considering several known Smart Contract Attacks & known issues.

- **Overflows and underflows**

An overflow happens when the limit of the type variable `uint256`, 2^{256} , is exceeded. What happens is that the value resets to zero instead of incrementing more.

For instance, if we want to assign a value to a `uint` bigger than 2^{256} it will simply go to 0—this is dangerous.

On the other hand, an underflow happens when you try to subtract 0 minus a number bigger than 0. For example, if you subtract $0 - 1$ the result will be 2^{256} instead of -1 .

This is quite dangerous. This contract DOES NOT check for overflows and underflows using [OpenZeppelin's SafeMath](#) for overflow and underflow protection. *However, starting 0.8.x solidity compilers have built in overflow checks.*

- **Reentrancy Attack**

One of the major dangers of [calling external contracts](#) is that they can take over the control flow, and make changes to your data that the calling function wasn't expecting. This class of bug can take many forms, and both of the major bugs that led to the DAO's collapse were bugs of this sort.

This smart contract does not include external calls for the most part, and the relayer includes external calls to the DLYCOP token contract which is trusted, and thus *is not found vulnerable* to re-entrancy attack.

- **Replay attack**

The replay attack consists of making a transaction on one blockchain like the original Ethereum's blockchain and then repeating it on another blockchain like the Ethereum's classic blockchain. The ether is transferred like a normal



transaction from a blockchain to another. Though it's no longer a problem because since the version 1.5.3 of *Geth* and 1.4.4 of *Parity* both implement the [attack protection EIP 155 by Vitalik Buterin](#).

So the users who use the contract depend on their own ability to be updated with those programs to keep themselves secure.

- **Short address attack**

This attack affects ERC20 tokens and consists of the following:

A user creates an Ethereum wallet with a trailing 0, which is not hard because it's only a digit. For instance: 0xiofa8d97756as7df5sd8f75g8675ds8gsdg0 (invalid address for discussion purpose)

Then he buys tokens by removing the last zero:

Buy 1000 tokens from account 0xiofa8d97756as7df5sd8f75g8675ds8gsdg. If the contract has enough amount of tokens and the buy function doesn't check the length of the address of the sender, the Ethereum's virtual machine will just add zeroes to the transaction until the address is complete.

The virtual machine will return 256000 for each 1000 tokens bought. This is a bug of the virtual machine.

Here is a [fix for short address attacks](#)

```
modifier onlyPayloadSize(uint size) {
    assert(msg.data.length >= size + 4);
    _;
}
function transfer(address _to, uint256 _value) onlyPayloadSize(2 * 32) {
    // do stuff
}
```

Whether or not it is appropriate for token contracts to mitigate the short-address attack is a contentious issue among smart-contract developers. Many, including those behind the OpenZeppelin project, have explicitly chosen not to do so. Blockchain Consilium doesn't consider short address attack an issue of the smart contract at the token level.

This contract **does not** implement an `onlyPayloadSize(uint numwords)` modifier for `transfer`, `transferFrom`, `approve`, `increaseAllowance`, and `decreaseAllowance` functions, it probably assumes that checks for short address attacks are handled at a higher layer (which generally are), and since the `onlyPayloadSize()` modifier [started causing some bugs restricting the flexibility](#) of the smart contracts, it's alright not to check for short address attacks at the Token Contract level to allow for some more flexibility for dAPP coding, but the checks for short address attacks must be done at some layer of coding (e.g. for buys and sells, the exchange can do it - almost all well-



known exchanges check for short address attacks), this contract *does not prevent short address attack*, so the *checks for short address attack must be done while buying or selling or coding a DAPP using DLYCOP where necessary*.

You can read more about the attack here: [ERC20 Short Address Attacks](#).

- **Approval Double-spend**

ERC20 Standard allows users to approve other users to manage their tokens, or spend tokens from their account till a certain amount, by setting the user's allowance with the standard `approve` function, then the allowed user may use `transferFrom` to spend the allowed tokens.

Hypothetically, given a situation where Alice approves Bob to spend 100 Tokens from her account, and if Alice needs to adjust the allowance to allow Bob to spend 20 more tokens, normally – she'd check Bob's allowance (100 currently) and start a new `approve` transaction allowing Bob to spend a total of 120 Tokens instead of 100 Tokens.

Now, if Bob is monitoring the Transaction pool, and as soon as he observes new transaction from Alice approving more amount, he may send a `transferFrom` transaction spending 100 Tokens from Alice's account with higher gas price and do all the required effort to get his spend transaction mined before Alice's new approve transaction.

Now Bob has already spent 100 Tokens, and given Alice's approve transaction is mined, Bob's allowance is set to 120 Tokens, this would allow Bob to spend a total of $100 + 120 = 220$ Tokens from Alice's account instead of the allowed 120 Tokens. This exploit situation is known as Approval Double-Spend Attack.

A potential solution to minimize these instances would be to set the non-zero allowance to 0 before setting it to any other amount.

It's possible for approve to enforce this behaviour without interface changes in the ERC20 specification:

```
if ((_value != 0) && (approved[msg.sender][_spender] != 0)) return false;
```

However, this is just an attempt to modify user behaviour. If the user does attempt to change from one non-zero value to another, the double spend might still happen, since the attacker may set the value to zero by already spending all the previously allowed value before the user's new approval transaction.

If desired, a non-standard function can be added to minimize hassle for users. The issue can be fixed with minimal inconvenience by taking a change value rather than a replacement value:

```
function increaseAllowance (address _spender, uint256 _addedValue)
returns (bool success) {
    uint oldValue = approved[msg.sender][_spender];
    approved[msg.sender][_spender] = safeAdd(oldValue, _addedValue);
    return true;
}
```

Even if this function is added, it's important to keep the original for compatibility with the ERC20 specification.

Likely impact of this bug is low for most situations. This contract **DOES** implement an `increaseAllowance` and a `decreaseAllowance` function, *thus it does apply mitigation for approval double-spend attack*.

For more, see this discussion on GitHub:

<https://github.com/ethereum/EIPs/issues/20#issuecomment263524729>

• Accidental Token Loss

- Token Smart Contracts should prevent transferring tokens to the token smart contract address if there's no good reason to not prevent, or if there's no way to take out tokens held by the token smart contract. The DLYCOP smart contract *does not* prevent transferring of DLYCOP to DLYCOP smart contract address. If someone accidentally sends DLYCOP to the DLYCOP smart contract, their DLYCOP will be locked up there forever with no way to take them out.

This can be prevented by not allowing transfer to token smart contract address.

- One more issue is when other ERC20 Tokens are transferred to the DLYCOP smart contract, there would be no way to take them out, and this can be solved by implementing the "Any Token Transfer" improvement suggestion.

• Sybil attacks

In a Sybil attack, the attacker subverts the reputation system of a network service by creating a large number of pseudonymous identities and uses them to gain a disproportionately large influence.



Normally in ERC20 & DeFi smart contracts, sybil attacks are related to voting power amplification where a user may use their vote more than once in case of governance tokens.

This smart contract does not implement inbuilt governance logic, thus it is not found vulnerable to sybil attacks.

Issues Found

High Severity Issues

No high severity issues were found in the smart contract.

Moderate Severity Issues

No moderate severity issues were found in the smart contract.

Low Severity Issues

1. DLYCOP *will be lost* / locked up with no way to take out, if sent to DLYCOP smart contract.
FIX: `require(recipient != address(this));` in `transfer` and `transferFrom` functions.
2. Any other ERC20 Tokens if sent to DLYCOP smart contract address, will be locked up forever with no way to take out
FIX: See the *"Any Token Transfer"* in improvements suggestions.

Both of the above issues are subjective, they can only happen if the user sends the tokens to the token smart contract addresses, though many times the user isn't aware or mistakenly copies wrong contract address, once sent, the tokens cannot be recovered if there's no way to take them out.

Informational Observations

- The Relayer is upgradeable, and the token is mintable with uncapped supply, it is recommended to keep the private keys of admin accounts secure – if a malicious actor gets access to the private keys, or gets access to the mint function directly or indirectly – it might be possible for them to mint a large number of tokens that can potentially allow them to steal significant amount of liquidity and market cap of the token, or to conduct a phishing exploit.
- In `DailyCopTokenChild.sol` at L#18 the ``_layer1Supply`` variable isn't assigned an initial value, this may not be an issue if the total supply is minted on



Polygon and then bridged to Ethereum, however if the total supply on L1 becomes inconsistent with this variable, chances are that `deposit` function will not behave properly due to potential underflow / overflow of `layer1Supply` variable, recommended to double check.

- The `hardhat/console.sol` import is not used, if it is not meant to be used it is recommended to remove unused imports.
- Recently it has been observed that several DeFi projects are launching their Tokens by adding significant liquidity to their Token's Uniswap pools, recently such launches, especially with significant amount of initial liquidity, are exploited by trading bots which intend to hoard up tokens by buying in large amounts – which pushes the token price up by a lot in a very short duration of time, and then after legit buyers purchase the tokens, the bots slowly sell the tokens (normally at a much higher price than the price these bots bought the tokens for). If the team is considering a Uniswap liquidity pool launch, with significant amount of initial liquidity, it is recommended to have non-standard time limited whitelisting structure in place to whitelist Ethereum accounts of legit buyers for an initial launch duration, with limited number of initial tokens to be purchased per account, which can potentially reduce the number of tokens purchased by bots during first few hours of the Uniswap liquidity pool launch.

This Token Smart Contract does not implement a non-standard time limited whitelist structure, though this is not required by the ERC20 Specification, it is a recommended non-standard solution to counter Uniswap bots. There may be other solutions out there as well. Whatever the solution the team and community choose to move forward with, it is highly recommended to have a strategy in place to counter trading bots for limited initial duration of a liquidity pool launch – if the team is considering launching Uniswap liquidity pool with a significant amount of initial liquidity.

Improvement Suggestions

We suggest adding two more optional features.

- **Ability to Reclaim Tokens transferred to Token Smart contract**

If someone inadvertently sends ERC20 Tokens to a wrong contract address, and the contract does not have any mechanism to call `transfer` on an arbitrary token contract, the smart contract will never be able to claim any tokens held by itself and those tokens will be locked up in the smart contract forever.

This can be avoided by adding a non-standard function to call `transfer` on an arbitrary token contract. That way, if someone inadvertently sends any ERC20 Tokens to this Token Contract, the owner will be able to call this function to transfer those tokens out of the contract.

This can be done by adding the following (or similar) function to the DLYCOP contract.

```
function transferAnyERC20(address _tokenAddress, address _to, uint _amount)
    public onlyOwner {
        IERC20(_tokenAddress).transfer(_to, _amount);
    }
```

Appendix

Smart Contract Summary

Relayer

- Contract IERC20Permit
 - From IERC20Permit
 - DOMAIN_SEPARATOR() (external)
 - nonces(address) (external)
 - permit(address,address,uint256,uint256,uint8,bytes32,bytes32) (external)
- Contract IERC20
 - From IERC20
 - allowance(address,address) (external)
 - approve(address,uint256) (external)
 - balanceOf(address) (external)
 - totalSupply() (external)
 - transfer(address,uint256) (external)
 - transferFrom(address,address,uint256) (external)
- Contract Initializable
- Contract IERC165Upgradeable
 - From IERC165Upgradeable
 - supportsInterface(bytes4) (external)
- Contract ERC165Upgradeable (Upgradeable)
 - From ERC165Upgradeable
 - __ERC165_init() (internal)
 - __ERC165_init_unchained() (internal)
 - supportsInterface(bytes4) (public)
- Contract StringsUpgradeable (Most derived contract)
 - From StringsUpgradeable
 - toHexString(uint256) (internal)



- toHexString(uint256,uint256) (internal)
- toString(uint256) (internal)
- Contract ContextUpgradeable (Upgradeable)
 - From ContextUpgradeable
 - __Context_init() (internal)
 - __Context_init_unchained() (internal)
 - _msgData() (internal)
 - _msgSender() (internal)
- Contract EnumerableSetUpgradeable (Most derived contract)
 - From EnumerableSetUpgradeable
 - _add(EnumerableSetUpgradeable.Set,bytes32) (private)
 - _at(EnumerableSetUpgradeable.Set,uint256) (private)
 - _contains(EnumerableSetUpgradeable.Set,bytes32) (private)
 - _length(EnumerableSetUpgradeable.Set) (private)
 - _remove(EnumerableSetUpgradeable.Set,bytes32) (private)
 - add(EnumerableSetUpgradeable.AddressSet,address) (internal)
 - add(EnumerableSetUpgradeable.Bytes32Set,bytes32) (internal)
 - add(EnumerableSetUpgradeable.UintSet,uint256) (internal)
 - at(EnumerableSetUpgradeable.AddressSet,uint256) (internal)
 - at(EnumerableSetUpgradeable.Bytes32Set,uint256) (internal)
 - at(EnumerableSetUpgradeable.UintSet,uint256) (internal)
 - contains(EnumerableSetUpgradeable.AddressSet,address) (internal)
 - contains(EnumerableSetUpgradeable.Bytes32Set,bytes32) (internal)
 - contains(EnumerableSetUpgradeable.UintSet,uint256) (internal)
 - length(EnumerableSetUpgradeable.AddressSet) (internal)
 - length(EnumerableSetUpgradeable.Bytes32Set) (internal)
 - length(EnumerableSetUpgradeable.UintSet) (internal)
 - remove(EnumerableSetUpgradeable.AddressSet,address) (internal)
 - remove(EnumerableSetUpgradeable.Bytes32Set,bytes32) (internal)
 - remove(EnumerableSetUpgradeable.UintSet,uint256) (internal)
- Contract IAccessControlUpgradeable



- From IAccessControlUpgradeable
 - getRoleAdmin(bytes32) (external)
 - grantRole(bytes32,address) (external)
 - hasRole(bytes32,address) (external)
 - renounceRole(bytes32,address) (external)
 - revokeRole(bytes32,address) (external)
- Contract AccessControlUpgradeable (Upgradeable)
 - From ERC165Upgradeable
 - __ERC165_init() (internal)
 - __ERC165_init_unchained() (internal)
 - From ContextUpgradeable
 - __Context_init() (internal)
 - __Context_init_unchained() (internal)
 - _msgData() (internal)
 - _msgSender() (internal)
 - From AccessControlUpgradeable
 - __AccessControl_init() (internal)
 - __AccessControl_init_unchained() (internal)
 - _checkRole(bytes32,address) (internal)
 - _grantRole(bytes32,address) (private)
 - _revokeRole(bytes32,address) (private)
 - _setRoleAdmin(bytes32,bytes32) (internal)
 - _setupRole(bytes32,address) (internal)
 - getRoleAdmin(bytes32) (public)
 - grantRole(bytes32,address) (public)
 - hasRole(bytes32,address) (public)
 - renounceRole(bytes32,address) (public)
 - revokeRole(bytes32,address) (public)
 - supportsInterface(bytes4) (public)
- Contract IDailyToken (Most derived contract)
 - From IERC20
 - allowance(address,address) (external)
 - approve(address,uint256) (external)
 - balanceOf(address) (external)
 - totalSupply() (external)



- transfer(address,uint256) (external)
 - transferFrom(address,address,uint256) (external)
- From IERC20Permit
 - DOMAIN_SEPARATOR() (external)
 - nonces(address) (external)
 - permit(address,address,uint256,uint256,uint8,bytes32,bytes32) (external)
- From IDailyToken
 - layer1Supply() (external)
- Contract IAccessControlEnumerableUpgradeable
 - From IAccessControlEnumerableUpgradeable
 - getRoleMember(bytes32,uint256) (external)
 - getRoleMemberCount(bytes32) (external)
- Contract AccessControlEnumerableUpgradeable (Upgradeable)
 - From AccessControlUpgradeable
 - __AccessControl_init() (internal)
 - __AccessControl_init_unchained() (internal)
 - _checkRole(bytes32,address) (internal)
 - _grantRole(bytes32,address) (private)
 - _revokeRole(bytes32,address) (private)
 - _setRoleAdmin(bytes32,bytes32) (internal)
 - getRoleAdmin(bytes32) (public)
 - hasRole(bytes32,address) (public)
 - From ERC165Upgradeable
 - __ERC165_init() (internal)
 - __ERC165_init_unchained() (internal)
 - From ContextUpgradeable
 - __Context_init() (internal)
 - __Context_init_unchained() (internal)
 - _msgData() (internal)
 - _msgSender() (internal)
 - From AccessControlEnumerableUpgradeable
 - __AccessControlEnumerable_init() (internal)
 - __AccessControlEnumerable_init_unchained() (internal)
 - _setupRole(bytes32,address) (internal)
 - getRoleMember(bytes32,uint256) (public)



- `getRoleMemberCount(bytes32)` (public)
 - `grantRole(bytes32,address)` (public)
 - `renounceRole(bytes32,address)` (public)
 - `revokeRole(bytes32,address)` (public)
 - `supportsInterface(bytes4)` (public)
- Contract console (Most derived contract)
 - From console
 - `_sendLogPayload(bytes)` (private)
 - `log()` (internal)
 - `log(address)` (internal)
 - `log(address,address)` (internal)
 - `log(address,address,address)` (internal)
 - `log(address,address,address,address)` (internal)
 - `log(address,address,address,bool)` (internal)
 - `log(address,address,address,string)` (internal)
 - `log(address,address,address,uint256)` (internal)
 - `log(address,address,bool)` (internal)
 - `log(address,address,bool,address)` (internal)
 - `log(address,address,bool,bool)` (internal)
 - `log(address,address,bool,string)` (internal)
 - `log(address,address,bool,uint256)` (internal)
 - `log(address,address,string)` (internal)
 - `log(address,address,string,address)` (internal)
 - `log(address,address,string,bool)` (internal)
 - `log(address,address,string,string)` (internal)
 - `log(address,address,string,uint256)` (internal)
 - `log(address,address,uint256)` (internal)
 - `log(address,address,uint256,address)` (internal)
 - `log(address,address,uint256,bool)` (internal)
 - `log(address,address,uint256,string)` (internal)
 - `log(address,address,uint256,uint256)` (internal)
 - `log(address,bool)` (internal)
 - `log(address,bool,address)` (internal)
 - `log(address,bool,address,address)` (internal)
 - `log(address,bool,address,bool)` (internal)
 - `log(address,bool,address,string)` (internal)



- log(address,bool,address,uint256) (internal)
- log(address,bool,bool) (internal)
- log(address,bool,bool,address) (internal)
- log(address,bool,bool,bool) (internal)
- log(address,bool,bool,string) (internal)
- log(address,bool,bool,uint256) (internal)
- log(address,bool,string) (internal)
- log(address,bool,string,address) (internal)
- log(address,bool,string,bool) (internal)
- log(address,bool,string,string) (internal)
- log(address,bool,string,uint256) (internal)
- log(address,bool,uint256) (internal)
- log(address,bool,uint256,address) (internal)
- log(address,bool,uint256,bool) (internal)
- log(address,bool,uint256,string) (internal)
- log(address,bool,uint256,uint256) (internal)
- log(address,string) (internal)
- log(address,string,address) (internal)
- log(address,string,address,address) (internal)
- log(address,string,address,bool) (internal)
- log(address,string,address,string) (internal)
- log(address,string,address,uint256) (internal)
- log(address,string,bool) (internal)
- log(address,string,bool,address) (internal)
- log(address,string,bool,bool) (internal)
- log(address,string,bool,string) (internal)
- log(address,string,bool,uint256) (internal)
- log(address,string,string) (internal)
- log(address,string,string,address) (internal)
- log(address,string,string,bool) (internal)
- log(address,string,string,string) (internal)
- log(address,string,string,uint256) (internal)
- log(address,string,uint256) (internal)
- log(address,string,uint256,address) (internal)
- log(address,string,uint256,bool) (internal)
- log(address,string,uint256,string) (internal)



- `log(address,string,uint256,uint256)` (internal)
- `log(address,uint256)` (internal)
- `log(address,uint256,address)` (internal)
- `log(address,uint256,address,address)` (internal)
- `log(address,uint256,address,bool)` (internal)
- `log(address,uint256,address,string)` (internal)
- `log(address,uint256,address,uint256)` (internal)
- `log(address,uint256,bool)` (internal)
- `log(address,uint256,bool,address)` (internal)
- `log(address,uint256,bool,bool)` (internal)
- `log(address,uint256,bool,string)` (internal)
- `log(address,uint256,bool,uint256)` (internal)
- `log(address,uint256,string)` (internal)
- `log(address,uint256,string,address)` (internal)
- `log(address,uint256,string,bool)` (internal)
- `log(address,uint256,string,string)` (internal)
- `log(address,uint256,string,uint256)` (internal)
- `log(address,uint256,uint256)` (internal)
- `log(address,uint256,uint256,address)` (internal)
- `log(address,uint256,uint256,bool)` (internal)
- `log(address,uint256,uint256,string)` (internal)
- `log(address,uint256,uint256,uint256)` (internal)
- `log(bool)` (internal)
- `log(bool,address)` (internal)
- `log(bool,address,address)` (internal)
- `log(bool,address,address,address)` (internal)
- `log(bool,address,address,bool)` (internal)
- `log(bool,address,address,string)` (internal)
- `log(bool,address,address,uint256)` (internal)
- `log(bool,address,bool)` (internal)
- `log(bool,address,bool,address)` (internal)
- `log(bool,address,bool,bool)` (internal)
- `log(bool,address,bool,string)` (internal)
- `log(bool,address,bool,uint256)` (internal)
- `log(bool,address,string)` (internal)
- `log(bool,address,string,address)` (internal)



- log(bool,address,string,bool) (internal)
- log(bool,address,string,string) (internal)
- log(bool,address,string,uint256) (internal)
- log(bool,address,uint256) (internal)
- log(bool,address,uint256,address) (internal)
- log(bool,address,uint256,bool) (internal)
- log(bool,address,uint256,string) (internal)
- log(bool,address,uint256,uint256) (internal)
- log(bool,bool) (internal)
- log(bool,bool,address) (internal)
- log(bool,bool,address,address) (internal)
- log(bool,bool,address,bool) (internal)
- log(bool,bool,address,string) (internal)
- log(bool,bool,address,uint256) (internal)
- log(bool,bool,bool) (internal)
- log(bool,bool,bool,address) (internal)
- log(bool,bool,bool,bool) (internal)
- log(bool,bool,bool,string) (internal)
- log(bool,bool,bool,uint256) (internal)
- log(bool,bool,string) (internal)
- log(bool,bool,string,address) (internal)
- log(bool,bool,string,bool) (internal)
- log(bool,bool,string,string) (internal)
- log(bool,bool,string,uint256) (internal)
- log(bool,bool,uint256) (internal)
- log(bool,bool,uint256,address) (internal)
- log(bool,bool,uint256,bool) (internal)
- log(bool,bool,uint256,string) (internal)
- log(bool,bool,uint256,uint256) (internal)
- log(bool,string) (internal)
- log(bool,string,address) (internal)
- log(bool,string,address,address) (internal)
- log(bool,string,address,bool) (internal)
- log(bool,string,address,string) (internal)
- log(bool,string,address,uint256) (internal)
- log(bool,string,bool) (internal)



- log(bool,string,bool,address) (internal)
- log(bool,string,bool,bool) (internal)
- log(bool,string,bool,string) (internal)
- log(bool,string,bool,uint256) (internal)
- log(bool,string,string) (internal)
- log(bool,string,string,address) (internal)
- log(bool,string,string,bool) (internal)
- log(bool,string,string,string) (internal)
- log(bool,string,string,uint256) (internal)
- log(bool,string,uint256) (internal)
- log(bool,string,uint256,address) (internal)
- log(bool,string,uint256,bool) (internal)
- log(bool,string,uint256,string) (internal)
- log(bool,string,uint256,uint256) (internal)
- log(bool,uint256) (internal)
- log(bool,uint256,address) (internal)
- log(bool,uint256,address,address) (internal)
- log(bool,uint256,address,bool) (internal)
- log(bool,uint256,address,string) (internal)
- log(bool,uint256,address,uint256) (internal)
- log(bool,uint256,bool) (internal)
- log(bool,uint256,bool,address) (internal)
- log(bool,uint256,bool,bool) (internal)
- log(bool,uint256,bool,string) (internal)
- log(bool,uint256,bool,uint256) (internal)
- log(bool,uint256,string) (internal)
- log(bool,uint256,string,address) (internal)
- log(bool,uint256,string,bool) (internal)
- log(bool,uint256,string,string) (internal)
- log(bool,uint256,string,uint256) (internal)
- log(bool,uint256,uint256) (internal)
- log(bool,uint256,uint256,address) (internal)
- log(bool,uint256,uint256,bool) (internal)
- log(bool,uint256,uint256,string) (internal)
- log(bool,uint256,uint256,uint256) (internal)
- log(string) (internal)



- `log(string,address)` (internal)
- `log(string,address,address)` (internal)
- `log(string,address,address,address)` (internal)
- `log(string,address,address,bool)` (internal)
- `log(string,address,address,string)` (internal)
- `log(string,address,address,uint256)` (internal)
- `log(string,address,bool)` (internal)
- `log(string,address,bool,address)` (internal)
- `log(string,address,bool,bool)` (internal)
- `log(string,address,bool,string)` (internal)
- `log(string,address,bool,uint256)` (internal)
- `log(string,address,string)` (internal)
- `log(string,address,string,address)` (internal)
- `log(string,address,string,bool)` (internal)
- `log(string,address,string,string)` (internal)
- `log(string,address,string,uint256)` (internal)
- `log(string,address,uint256)` (internal)
- `log(string,address,uint256,address)` (internal)
- `log(string,address,uint256,bool)` (internal)
- `log(string,address,uint256,string)` (internal)
- `log(string,address,uint256,uint256)` (internal)
- `log(string,bool)` (internal)
- `log(string,bool,address)` (internal)
- `log(string,bool,address,address)` (internal)
- `log(string,bool,address,bool)` (internal)
- `log(string,bool,address,string)` (internal)
- `log(string,bool,address,uint256)` (internal)
- `log(string,bool,bool)` (internal)
- `log(string,bool,bool,address)` (internal)
- `log(string,bool,bool,bool)` (internal)
- `log(string,bool,bool,string)` (internal)
- `log(string,bool,bool,uint256)` (internal)
- `log(string,bool,string)` (internal)
- `log(string,bool,string,address)` (internal)
- `log(string,bool,string,bool)` (internal)
- `log(string,bool,string,string)` (internal)



- `log(string,bool,string,uint256)` (internal)
- `log(string,bool,uint256)` (internal)
- `log(string,bool,uint256,address)` (internal)
- `log(string,bool,uint256,bool)` (internal)
- `log(string,bool,uint256,string)` (internal)
- `log(string,bool,uint256,uint256)` (internal)
- `log(string,string)` (internal)
- `log(string,string,address)` (internal)
- `log(string,string,address,address)` (internal)
- `log(string,string,address,bool)` (internal)
- `log(string,string,address,string)` (internal)
- `log(string,string,address,uint256)` (internal)
- `log(string,string,bool)` (internal)
- `log(string,string,bool,address)` (internal)
- `log(string,string,bool,bool)` (internal)
- `log(string,string,bool,string)` (internal)
- `log(string,string,bool,uint256)` (internal)
- `log(string,string,string)` (internal)
- `log(string,string,string,address)` (internal)
- `log(string,string,string,bool)` (internal)
- `log(string,string,string,string)` (internal)
- `log(string,string,string,uint256)` (internal)
- `log(string,string,uint256)` (internal)
- `log(string,string,uint256,address)` (internal)
- `log(string,string,uint256,bool)` (internal)
- `log(string,string,uint256,string)` (internal)
- `log(string,string,uint256,uint256)` (internal)
- `log(string,uint256)` (internal)
- `log(string,uint256,address)` (internal)
- `log(string,uint256,address,address)` (internal)
- `log(string,uint256,address,bool)` (internal)
- `log(string,uint256,address,string)` (internal)
- `log(string,uint256,address,uint256)` (internal)
- `log(string,uint256,bool)` (internal)
- `log(string,uint256,bool,address)` (internal)
- `log(string,uint256,bool,bool)` (internal)



- `log(string,uint256,bool,string)` (internal)
- `log(string,uint256,bool,uint256)` (internal)
- `log(string,uint256,string)` (internal)
- `log(string,uint256,string,address)` (internal)
- `log(string,uint256,string,bool)` (internal)
- `log(string,uint256,string,string)` (internal)
- `log(string,uint256,string,uint256)` (internal)
- `log(string,uint256,uint256)` (internal)
- `log(string,uint256,uint256,address)` (internal)
- `log(string,uint256,uint256,bool)` (internal)
- `log(string,uint256,uint256,string)` (internal)
- `log(string,uint256,uint256,uint256)` (internal)
- `log(uint256)` (internal)
- `log(uint256,address)` (internal)
- `log(uint256,address,address)` (internal)
- `log(uint256,address,address,address)` (internal)
- `log(uint256,address,address,bool)` (internal)
- `log(uint256,address,address,string)` (internal)
- `log(uint256,address,address,uint256)` (internal)
- `log(uint256,address,bool)` (internal)
- `log(uint256,address,bool,address)` (internal)
- `log(uint256,address,bool,bool)` (internal)
- `log(uint256,address,bool,string)` (internal)
- `log(uint256,address,bool,uint256)` (internal)
- `log(uint256,address,string)` (internal)
- `log(uint256,address,string,address)` (internal)
- `log(uint256,address,string,bool)` (internal)
- `log(uint256,address,string,string)` (internal)
- `log(uint256,address,string,uint256)` (internal)
- `log(uint256,address,uint256)` (internal)
- `log(uint256,address,uint256,address)` (internal)
- `log(uint256,address,uint256,bool)` (internal)
- `log(uint256,address,uint256,string)` (internal)
- `log(uint256,address,uint256,uint256)` (internal)
- `log(uint256,bool)` (internal)
- `log(uint256,bool,address)` (internal)



- `log(uint256,bool,address,address)` (internal)
- `log(uint256,bool,address,bool)` (internal)
- `log(uint256,bool,address,string)` (internal)
- `log(uint256,bool,address,uint256)` (internal)
- `log(uint256,bool,bool)` (internal)
- `log(uint256,bool,bool,address)` (internal)
- `log(uint256,bool,bool,bool)` (internal)
- `log(uint256,bool,bool,string)` (internal)
- `log(uint256,bool,bool,uint256)` (internal)
- `log(uint256,bool,string)` (internal)
- `log(uint256,bool,string,address)` (internal)
- `log(uint256,bool,string,bool)` (internal)
- `log(uint256,bool,string,string)` (internal)
- `log(uint256,bool,string,uint256)` (internal)
- `log(uint256,bool,uint256)` (internal)
- `log(uint256,bool,uint256,address)` (internal)
- `log(uint256,bool,uint256,bool)` (internal)
- `log(uint256,bool,uint256,string)` (internal)
- `log(uint256,bool,uint256,uint256)` (internal)
- `log(uint256,string)` (internal)
- `log(uint256,string,address)` (internal)
- `log(uint256,string,address,address)` (internal)
- `log(uint256,string,address,bool)` (internal)
- `log(uint256,string,address,string)` (internal)
- `log(uint256,string,address,uint256)` (internal)
- `log(uint256,string,bool)` (internal)
- `log(uint256,string,bool,address)` (internal)
- `log(uint256,string,bool,bool)` (internal)
- `log(uint256,string,bool,string)` (internal)
- `log(uint256,string,bool,uint256)` (internal)
- `log(uint256,string,string)` (internal)
- `log(uint256,string,string,address)` (internal)
- `log(uint256,string,string,bool)` (internal)
- `log(uint256,string,string,string)` (internal)
- `log(uint256,string,string,uint256)` (internal)
- `log(uint256,string,uint256)` (internal)



- `log(uint256,string,uint256,address)` (internal)
- `log(uint256,string,uint256,bool)` (internal)
- `log(uint256,string,uint256,string)` (internal)
- `log(uint256,string,uint256,uint256)` (internal)
- `log(uint256,uint256)` (internal)
- `log(uint256,uint256,address)` (internal)
- `log(uint256,uint256,address,address)` (internal)
- `log(uint256,uint256,address,bool)` (internal)
- `log(uint256,uint256,address,string)` (internal)
- `log(uint256,uint256,address,uint256)` (internal)
- `log(uint256,uint256,bool)` (internal)
- `log(uint256,uint256,bool,address)` (internal)
- `log(uint256,uint256,bool,bool)` (internal)
- `log(uint256,uint256,bool,string)` (internal)
- `log(uint256,uint256,bool,uint256)` (internal)
- `log(uint256,uint256,string)` (internal)
- `log(uint256,uint256,string,address)` (internal)
- `log(uint256,uint256,string,bool)` (internal)
- `log(uint256,uint256,string,string)` (internal)
- `log(uint256,uint256,string,uint256)` (internal)
- `log(uint256,uint256,uint256)` (internal)
- `log(uint256,uint256,uint256,address)` (internal)
- `log(uint256,uint256,uint256,bool)` (internal)
- `log(uint256,uint256,uint256,string)` (internal)
- `log(uint256,uint256,uint256,uint256)` (internal)
- `logAddress(address)` (internal)
- `logBool(bool)` (internal)
- `logBytes(bytes)` (internal)
- `logBytes1(bytes1)` (internal)
- `logBytes10(bytes10)` (internal)
- `logBytes11(bytes11)` (internal)
- `logBytes12(bytes12)` (internal)
- `logBytes13(bytes13)` (internal)
- `logBytes14(bytes14)` (internal)
- `logBytes15(bytes15)` (internal)
- `logBytes16(bytes16)` (internal)



- logBytes17(bytes17) (internal)
- logBytes18(bytes18) (internal)
- logBytes19(bytes19) (internal)
- logBytes2(bytes2) (internal)
- logBytes20(bytes20) (internal)
- logBytes21(bytes21) (internal)
- logBytes22(bytes22) (internal)
- logBytes23(bytes23) (internal)
- logBytes24(bytes24) (internal)
- logBytes25(bytes25) (internal)
- logBytes26(bytes26) (internal)
- logBytes27(bytes27) (internal)
- logBytes28(bytes28) (internal)
- logBytes29(bytes29) (internal)
- logBytes3(bytes3) (internal)
- logBytes30(bytes30) (internal)
- logBytes31(bytes31) (internal)
- logBytes32(bytes32) (internal)
- logBytes4(bytes4) (internal)
- logBytes5(bytes5) (internal)
- logBytes6(bytes6) (internal)
- logBytes7(bytes7) (internal)
- logBytes8(bytes8) (internal)
- logBytes9(bytes9) (internal)
- logInt(int256) (internal)
- logString(string) (internal)
- logUint(uint256) (internal)
- Contract Relayer (Upgradeable) (Most derived contract)
 - From AccessControlEnumerableUpgradeable
 - __AccessControlEnumerable_init() (internal)
 - __AccessControlEnumerable_init_unchained() (internal)
 - _setupRole(bytes32,address) (internal)
 - getRoleMember(bytes32,uint256) (public)
 - getRoleMemberCount(bytes32) (public)
 - grantRole(bytes32,address) (public)
 - renounceRole(bytes32,address) (public)



- revokeRole(bytes32,address) (public)
- supportsInterface(bytes4) (public)
- From AccessControlUpgradeable
 - __AccessControl_init() (internal)
 - __AccessControl_init_unchained() (internal)
 - _checkRole(bytes32,address) (internal)
 - _grantRole(bytes32,address) (private)
 - _revokeRole(bytes32,address) (private)
 - _setRoleAdmin(bytes32,bytes32) (internal)
 - getRoleAdmin(bytes32) (public)
 - hasRole(bytes32,address) (public)
- From ERC165Upgradeable
 - __ERC165_init() (internal)
 - __ERC165_init_unchained() (internal)
- From ContextUpgradeable
 - __Context_init() (internal)
 - __Context_init_unchained() (internal)
 - _msgData() (internal)
 - _msgSender() (internal)
- From Relayer
 - initialize(address,address,IDailyToken) (public)
 - transferWithPermit(address,address,uint256,uint256,uint8,bytes32,bytes32) (external)

DailyCopTokenRoot

- Contract ECDSA (Most derived contract)
 - From ECDSA
 - recover(bytes32,bytes) (internal)
 - recover(bytes32,bytes32,bytes32) (internal)
 - recover(bytes32,uint8,bytes32,bytes32) (internal)
 - toEthSignedMessageHash(bytes32) (internal)
 - toTypedDataHash(bytes32,bytes32) (internal)
- Contract IERC20
 - From IERC20
 - allowance(address,address) (external)
 - approve(address,uint256) (external)



- `balanceOf(address)` (external)
 - `totalSupply()` (external)
 - `transfer(address,uint256)` (external)
 - `transferFrom(address,address,uint256)` (external)
- Contract Context
 - From Context
 - `_msgData()` (internal)
 - `_msgSender()` (internal)
- Contract IERC20Metadata
 - From IERC20
 - `allowance(address,address)` (external)
 - `approve(address,uint256)` (external)
 - `balanceOf(address)` (external)
 - `totalSupply()` (external)
 - `transfer(address,uint256)` (external)
 - `transferFrom(address,address,uint256)` (external)
 - From IERC20Metadata
 - `decimals()` (external)
 - `name()` (external)
 - `symbol()` (external)
- Contract Counters (Most derived contract)
 - From Counters
 - `current(Counters.Counter)` (internal)
 - `decrement(Counters.Counter)` (internal)
 - `increment(Counters.Counter)` (internal)
 - `reset(Counters.Counter)` (internal)
- Contract EIP712
 - From EIP712
 - `_buildDomainSeparator(bytes32,bytes32,bytes32)` (private)
 - `_domainSeparatorV4()` (internal)
 - `_hashTypedDataV4(bytes32)` (internal)
 - `constructor(string,string)` (internal)
- Contract ERC20

- From Context
 - `_msgData()` (internal)
 - `_msgSender()` (internal)
- From ERC20
 - `_afterTokenTransfer(address,address,uint256)` (internal)
 - `_approve(address,address,uint256)` (internal)
 - `_beforeTokenTransfer(address,address,uint256)` (internal)
 - `_burn(address,uint256)` (internal)
 - `_mint(address,uint256)` (internal)
 - `_transfer(address,address,uint256)` (internal)
 - `allowance(address,address)` (public)
 - `approve(address,uint256)` (public)
 - `balanceOf(address)` (public)
 - `constructor(string,string)` (public)
 - `decimals()` (public)
 - `decreaseAllowance(address,uint256)` (public)
 - `increaseAllowance(address,uint256)` (public)
 - `name()` (public)
 - `symbol()` (public)
 - `totalSupply()` (public)
 - `transfer(address,uint256)` (public)
 - `transferFrom(address,address,uint256)` (public)
- Contract IERC20Permit
 - From IERC20Permit
 - `DOMAIN_SEPARATOR()` (external)
 - `nonces(address)` (external)
 - `permit(address,address,uint256,uint256,uint8,bytes32,bytes32)` (external)
- Contract IERC165
 - From IERC165
 - `supportsInterface(bytes4)` (external)
- Contract ERC165
 - From ERC165
 - `supportsInterface(bytes4)` (public)



- Contract Strings (Most derived contract)
 - From Strings
 - toHexString(uint256) (internal)
 - toHexString(uint256,uint256) (internal)
 - toString(uint256) (internal)
- Contract EnumerableSet (Most derived contract)
 - From EnumerableSet
 - _add(EnumerableSet.Set,bytes32) (private)
 - _at(EnumerableSet.Set,uint256) (private)
 - _contains(EnumerableSet.Set,bytes32) (private)
 - _length(EnumerableSet.Set) (private)
 - _remove(EnumerableSet.Set,bytes32) (private)
 - add(EnumerableSet.AddressSet,address) (internal)
 - add(EnumerableSet.Bytes32Set,bytes32) (internal)
 - add(EnumerableSet.UintSet,uint256) (internal)
 - at(EnumerableSet.AddressSet,uint256) (internal)
 - at(EnumerableSet.Bytes32Set,uint256) (internal)
 - at(EnumerableSet.UintSet,uint256) (internal)
 - contains(EnumerableSet.AddressSet,address) (internal)
 - contains(EnumerableSet.Bytes32Set,bytes32) (internal)
 - contains(EnumerableSet.UintSet,uint256) (internal)
 - length(EnumerableSet.AddressSet) (internal)
 - length(EnumerableSet.Bytes32Set) (internal)
 - length(EnumerableSet.UintSet) (internal)
 - remove(EnumerableSet.AddressSet,address) (internal)
 - remove(EnumerableSet.Bytes32Set,bytes32) (internal)
 - remove(EnumerableSet.UintSet,uint256) (internal)
- Contract IAccessControl
 - From IAccessControl
 - getRoleAdmin(bytes32) (external)
 - grantRole(bytes32,address) (external)
 - hasRole(bytes32,address) (external)
 - renounceRole(bytes32,address) (external)
 - revokeRole(bytes32,address) (external)



- Contract AccessControl
 - From Context
 - _msgData() (internal)
 - _msgSender() (internal)
 - From AccessControl
 - _checkRole(bytes32,address) (internal)
 - _grantRole(bytes32,address) (private)
 - _revokeRole(bytes32,address) (private)
 - _setRoleAdmin(bytes32,bytes32) (internal)
 - _setupRole(bytes32,address) (internal)
 - getRoleAdmin(bytes32) (public)
 - grantRole(bytes32,address) (public)
 - hasRole(bytes32,address) (public)
 - renounceRole(bytes32,address) (public)
 - revokeRole(bytes32,address) (public)
 - supportsInterface(bytes4) (public)
- Contract ERC20Permit
 - From EIP712
 - _buildDomainSeparator(bytes32,bytes32,bytes32) (private)
 - _domainSeparatorV4() (internal)
 - _hashTypedDataV4(bytes32) (internal)
 - constructor(string,string) (internal)
 - From ERC20
 - _afterTokenTransfer(address,address,uint256) (internal)
 - _approve(address,address,uint256) (internal)
 - _beforeTokenTransfer(address,address,uint256) (internal)
 - _burn(address,uint256) (internal)
 - _mint(address,uint256) (internal)
 - _transfer(address,address,uint256) (internal)
 - allowance(address,address) (public)
 - approve(address,uint256) (public)
 - balanceOf(address) (public)
 - decimals() (public)
 - decreaseAllowance(address,uint256) (public)
 - increaseAllowance(address,uint256) (public)
 - name() (public)



- symbol() (public)
 - totalSupply() (public)
 - transfer(address,uint256) (public)
 - transferFrom(address,address,uint256) (public)
- From Context
 - _msgData() (internal)
 - _msgSender() (internal)
- From ERC20Permit
 - DOMAIN_SEPARATOR() (external)
 - _useNonce(address) (internal)
 - constructor(string) (internal)
 - nonces(address) (public)
 - permit(address,address,uint256,uint256,uint8,bytes32,bytes32) (public)
- Contract IAccessControlEnumerable
 - From IAccessControlEnumerable
 - getRoleMember(bytes32,uint256) (external)
 - getRoleMemberCount(bytes32) (external)
- Contract AccessControlEnumerable
 - From AccessControl
 - _checkRole(bytes32,address) (internal)
 - _grantRole(bytes32,address) (private)
 - _revokeRole(bytes32,address) (private)
 - _setRoleAdmin(bytes32,bytes32) (internal)
 - getRoleAdmin(bytes32) (public)
 - hasRole(bytes32,address) (public)
 - From Context
 - _msgData() (internal)
 - _msgSender() (internal)
 - From AccessControlEnumerable
 - _setupRole(bytes32,address) (internal)
 - getRoleMember(bytes32,uint256) (public)
 - getRoleMemberCount(bytes32) (public)
 - grantRole(bytes32,address) (public)
 - renounceRole(bytes32,address) (public)
 - revokeRole(bytes32,address) (public)



- supportsInterface(bytes4) (public)
- Contract IMintableERC20
 - From IERC20
 - allowance(address,address) (external)
 - approve(address,uint256) (external)
 - balanceOf(address) (external)
 - totalSupply() (external)
 - transfer(address,uint256) (external)
 - transferFrom(address,address,uint256) (external)
 - From IMintableERC20
 - mint(address,uint256) (external)
- Contract DailyCopTokenRoot (Most derived contract)
 - From AccessControlEnumerable
 - _setupRole(bytes32,address) (internal)
 - getRoleMember(bytes32,uint256) (public)
 - getRoleMemberCount(bytes32) (public)
 - grantRole(bytes32,address) (public)
 - renounceRole(bytes32,address) (public)
 - revokeRole(bytes32,address) (public)
 - supportsInterface(bytes4) (public)
 - From AccessControl
 - _checkRole(bytes32,address) (internal)
 - _grantRole(bytes32,address) (private)
 - _revokeRole(bytes32,address) (private)
 - _setRoleAdmin(bytes32,bytes32) (internal)
 - getRoleAdmin(bytes32) (public)
 - hasRole(bytes32,address) (public)
 - From Context
 - _msgData() (internal)
 - _msgSender() (internal)
 - From ERC20Permit
 - DOMAIN_SEPARATOR() (external)
 - _useNonce(address) (internal)
 - constructor(string) (internal)
 - nonces(address) (public)



- permit(address,address,uint256,uint256,uint8,bytes32,bytes32) (public)
- From EIP712
 - _buildDomainSeparator(bytes32,bytes32,bytes32) (private)
 - _domainSeparatorV4() (internal)
 - _hashTypedDataV4(bytes32) (internal)
 - constructor(string,string) (internal)
- From ERC20
 - _afterTokenTransfer(address,address,uint256) (internal)
 - _approve(address,address,uint256) (internal)
 - _beforeTokenTransfer(address,address,uint256) (internal)
 - _burn(address,uint256) (internal)
 - _mint(address,uint256) (internal)
 - _transfer(address,address,uint256) (internal)
 - allowance(address,address) (public)
 - approve(address,uint256) (public)
 - balanceOf(address) (public)
 - decimals() (public)
 - decreaseAllowance(address,uint256) (public)
 - increaseAllowance(address,uint256) (public)
 - name() (public)
 - symbol() (public)
 - totalSupply() (public)
 - transfer(address,uint256) (public)
 - transferFrom(address,address,uint256) (public)
- From DailyCopTokenRoot
 - constructor(address) (public)
 - mint(address,uint256) (external)

DailyCopTokenChild

- Contract IERC20
 - From IERC20
 - allowance(address,address) (external)
 - approve(address,uint256) (external)
 - balanceOf(address) (external)
 - totalSupply() (external)
 - transfer(address,uint256) (external)



- `transferFrom(address,address,uint256)` (external)
- Contract Math (Most derived contract)
 - From Math
 - `average(uint256,uint256)` (internal)
 - `ceilDiv(uint256,uint256)` (internal)
 - `max(uint256,uint256)` (internal)
 - `min(uint256,uint256)` (internal)
- Contract Context
 - From Context
 - `_msgData()` (internal)
 - `_msgSender()` (internal)
- Contract IERC20Metadata
 - From IERC20
 - `allowance(address,address)` (external)
 - `approve(address,uint256)` (external)
 - `balanceOf(address)` (external)
 - `totalSupply()` (external)
 - `transfer(address,uint256)` (external)
 - `transferFrom(address,address,uint256)` (external)
 - From IERC20Metadata
 - `decimals()` (external)
 - `name()` (external)
 - `symbol()` (external)
- Contract Counters (Most derived contract)
 - From Counters
 - `current(Counters.Counter)` (internal)
 - `decrement(Counters.Counter)` (internal)
 - `increment(Counters.Counter)` (internal)
 - `reset(Counters.Counter)` (internal)
- Contract Arrays (Most derived contract)
 - From Arrays
 - `findUpperBound(uint256[],uint256)` (internal)



- Contract ERC20
 - From Context
 - `_msgData()` (internal)
 - `_msgSender()` (internal)
 - From ERC20
 - `_afterTokenTransfer(address,address,uint256)` (internal)
 - `_approve(address,address,uint256)` (internal)
 - `_beforeTokenTransfer(address,address,uint256)` (internal)
 - `_burn(address,uint256)` (internal)
 - `_mint(address,uint256)` (internal)
 - `_transfer(address,address,uint256)` (internal)
 - `allowance(address,address)` (public)
 - `approve(address,uint256)` (public)
 - `balanceOf(address)` (public)
 - `constructor(string,string)` (public)
 - `decimals()` (public)
 - `decreaseAllowance(address,uint256)` (public)
 - `increaseAllowance(address,uint256)` (public)
 - `name()` (public)
 - `symbol()` (public)
 - `totalSupply()` (public)
 - `transfer(address,uint256)` (public)
 - `transferFrom(address,address,uint256)` (public)
- Contract IERC165
 - From IERC165
 - `supportsInterface(bytes4)` (external)
- Contract ERC165
 - From ERC165
 - `supportsInterface(bytes4)` (public)
- Contract Strings (Most derived contract)
 - From Strings
 - `toHexString(uint256)` (internal)
 - `toHexString(uint256,uint256)` (internal)
 - `toString(uint256)` (internal)



- Contract EnumerableSet (Most derived contract)
 - From EnumerableSet
 - `_add(EnumerableSet.Set,bytes32)` (private)
 - `_at(EnumerableSet.Set,uint256)` (private)
 - `_contains(EnumerableSet.Set,bytes32)` (private)
 - `_length(EnumerableSet.Set)` (private)
 - `_remove(EnumerableSet.Set,bytes32)` (private)
 - `add(EnumerableSet.AddressSet,address)` (internal)
 - `add(EnumerableSet.Bytes32Set,bytes32)` (internal)
 - `add(EnumerableSet.UintSet,uint256)` (internal)
 - `at(EnumerableSet.AddressSet,uint256)` (internal)
 - `at(EnumerableSet.Bytes32Set,uint256)` (internal)
 - `at(EnumerableSet.UintSet,uint256)` (internal)
 - `contains(EnumerableSet.AddressSet,address)` (internal)
 - `contains(EnumerableSet.Bytes32Set,bytes32)` (internal)
 - `contains(EnumerableSet.UintSet,uint256)` (internal)
 - `length(EnumerableSet.AddressSet)` (internal)
 - `length(EnumerableSet.Bytes32Set)` (internal)
 - `length(EnumerableSet.UintSet)` (internal)
 - `remove(EnumerableSet.AddressSet,address)` (internal)
 - `remove(EnumerableSet.Bytes32Set,bytes32)` (internal)
 - `remove(EnumerableSet.UintSet,uint256)` (internal)
- Contract IAccessControl
 - From IAccessControl
 - `getRoleAdmin(bytes32)` (external)
 - `grantRole(bytes32,address)` (external)
 - `hasRole(bytes32,address)` (external)
 - `renounceRole(bytes32,address)` (external)
 - `revokeRole(bytes32,address)` (external)
- Contract AccessControl
 - From Context
 - `_msgData()` (internal)
 - `_msgSender()` (internal)
 - From AccessControl
 - `_checkRole(bytes32,address)` (internal)



- `_grantRole(bytes32,address)` (private)
 - `_revokeRole(bytes32,address)` (private)
 - `_setRoleAdmin(bytes32,bytes32)` (internal)
 - `_setupRole(bytes32,address)` (internal)
 - `getRoleAdmin(bytes32)` (public)
 - `grantRole(bytes32,address)` (public)
 - `hasRole(bytes32,address)` (public)
 - `renounceRole(bytes32,address)` (public)
 - `revokeRole(bytes32,address)` (public)
 - `supportsInterface(bytes4)` (public)
- Contract ECDSA (Most derived contract)
 - From ECDSA
 - `recover(bytes32,bytes)` (internal)
 - `recover(bytes32,bytes32,bytes32)` (internal)
 - `recover(bytes32,uint8,bytes32,bytes32)` (internal)
 - `toEthSignedMessageHash(bytes32)` (internal)
 - `toTypedDataHash(bytes32,bytes32)` (internal)
- Contract EIP712
 - From EIP712
 - `_buildDomainSeparator(bytes32,bytes32,bytes32)` (private)
 - `_domainSeparatorV4()` (internal)
 - `_hashTypedDataV4(bytes32)` (internal)
 - `constructor(string,string)` (internal)
- Contract IERC20Permit
 - From IERC20Permit
 - `DOMAIN_SEPARATOR()` (external)
 - `nonces(address)` (external)
 - `permit(address,address,uint256,uint256,uint8,bytes32,bytes32)` (external)
- Contract IChildToken
 - From IERC20
 - `allowance(address,address)` (external)
 - `approve(address,uint256)` (external)



- `balanceOf(address)` (external)
 - `totalSupply()` (external)
 - `transfer(address,uint256)` (external)
 - `transferFrom(address,address,uint256)` (external)
- From `ICildToken`
 - `deposit(address,bytes)` (external)
- Contract `ERC20Snapshot`
 - From `ERC20`
 - `_afterTokenTransfer(address,address,uint256)` (internal)
 - `_approve(address,address,uint256)` (internal)
 - `_burn(address,uint256)` (internal)
 - `_mint(address,uint256)` (internal)
 - `_transfer(address,address,uint256)` (internal)
 - `allowance(address,address)` (public)
 - `approve(address,uint256)` (public)
 - `balanceOf(address)` (public)
 - `constructor(string,string)` (public)
 - `decimals()` (public)
 - `decreaseAllowance(address,uint256)` (public)
 - `increaseAllowance(address,uint256)` (public)
 - `name()` (public)
 - `symbol()` (public)
 - `totalSupply()` (public)
 - `transfer(address,uint256)` (public)
 - `transferFrom(address,address,uint256)` (public)
 - From `Context`
 - `_msgData()` (internal)
 - `_msgSender()` (internal)
 - From `ERC20Snapshot`
 - `_beforeTokenTransfer(address,address,uint256)` (internal)
 - `_getCurrentSnapshotId()` (internal)
 - `_lastSnapshotId(uint256[])` (private)
 - `_snapshot()` (internal)
 - `_updateAccountSnapshot(address)` (private)
 - `_updateSnapshot(ERC20Snapshot.Snapshots,uint256)` (private)
 - `_updateTotalSupplySnapshot()` (private)



- `_valueAt(uint256,ERC20Snapshot.Snapshots)` (private)
 - `balanceOfAt(address,uint256)` (public)
 - `totalSupplyAt(uint256)` (public)
- Contract `IAccessControlEnumerable`
 - From `IAccessControlEnumerable`
 - `getRoleMember(bytes32,uint256)` (external)
 - `getRoleMemberCount(bytes32)` (external)
- Contract `AccessControlEnumerable`
 - From `AccessControl`
 - `_checkRole(bytes32,address)` (internal)
 - `_grantRole(bytes32,address)` (private)
 - `_revokeRole(bytes32,address)` (private)
 - `_setRoleAdmin(bytes32,bytes32)` (internal)
 - `getRoleAdmin(bytes32)` (public)
 - `hasRole(bytes32,address)` (public)
 - From `Context`
 - `_msgData()` (internal)
 - `_msgSender()` (internal)
 - From `AccessControlEnumerable`
 - `_setupRole(bytes32,address)` (internal)
 - `getRoleMember(bytes32,uint256)` (public)
 - `getRoleMemberCount(bytes32)` (public)
 - `grantRole(bytes32,address)` (public)
 - `renounceRole(bytes32,address)` (public)
 - `revokeRole(bytes32,address)` (public)
 - `supportsInterface(bytes4)` (public)
- Contract `ERC20Burnable`
 - From `ERC20`
 - `_afterTokenTransfer(address,address,uint256)` (internal)
 - `_approve(address,address,uint256)` (internal)
 - `_beforeTokenTransfer(address,address,uint256)` (internal)
 - `_burn(address,uint256)` (internal)
 - `_mint(address,uint256)` (internal)
 - `_transfer(address,address,uint256)` (internal)



- allowance(address,address) (public)
 - approve(address,uint256) (public)
 - balanceOf(address) (public)
 - constructor(string,string) (public)
 - decimals() (public)
 - decreaseAllowance(address,uint256) (public)
 - increaseAllowance(address,uint256) (public)
 - name() (public)
 - symbol() (public)
 - totalSupply() (public)
 - transfer(address,uint256) (public)
 - transferFrom(address,address,uint256) (public)
- From Context
 - _msgData() (internal)
 - _msgSender() (internal)
- From ERC20Burnable
 - burn(uint256) (public)
 - burnFrom(address,uint256) (public)
- Contract ERC20Permit
 - From EIP712
 - _buildDomainSeparator(bytes32,bytes32,bytes32) (private)
 - _domainSeparatorV4() (internal)
 - _hashTypedDataV4(bytes32) (internal)
 - constructor(string,string) (internal)
 - From ERC20
 - _afterTokenTransfer(address,address,uint256) (internal)
 - _approve(address,address,uint256) (internal)
 - _beforeTokenTransfer(address,address,uint256) (internal)
 - _burn(address,uint256) (internal)
 - _mint(address,uint256) (internal)
 - _transfer(address,address,uint256) (internal)
 - allowance(address,address) (public)
 - approve(address,uint256) (public)
 - balanceOf(address) (public)
 - decimals() (public)
 - decreaseAllowance(address,uint256) (public)



- `increaseAllowance(address,uint256)` (public)
 - `name()` (public)
 - `symbol()` (public)
 - `totalSupply()` (public)
 - `transfer(address,uint256)` (public)
 - `transferFrom(address,address,uint256)` (public)
- From Context
 - `_msgData()` (internal)
 - `_msgSender()` (internal)
- From ERC20Permit
 - `DOMAIN_SEPARATOR()` (external)
 - `_useNonce(address)` (internal)
 - `constructor(string)` (internal)
 - `nonces(address)` (public)
 - `permit(address,address,uint256,uint256,uint8,bytes32,bytes32)` (public)
- Contract console (Most derived contract)
 - From console
 - `_sendLogPayload(bytes)` (private)
 - `log()` (internal)
 - `log(address)` (internal)
 - `log(address,address)` (internal)
 - `log(address,address,address)` (internal)
 - `log(address,address,address,address)` (internal)
 - `log(address,address,address,bool)` (internal)
 - `log(address,address,address,string)` (internal)
 - `log(address,address,address,uint256)` (internal)
 - `log(address,address,bool)` (internal)
 - `log(address,address,bool,address)` (internal)
 - `log(address,address,bool,bool)` (internal)
 - `log(address,address,bool,string)` (internal)
 - `log(address,address,bool,uint256)` (internal)
 - `log(address,address,string)` (internal)
 - `log(address,address,string,address)` (internal)
 - `log(address,address,string,bool)` (internal)
 - `log(address,address,string,string)` (internal)



- log(address,address,string,uint256) (internal)
- log(address,address,uint256) (internal)
- log(address,address,uint256,address) (internal)
- log(address,address,uint256,bool) (internal)
- log(address,address,uint256,string) (internal)
- log(address,address,uint256,uint256) (internal)
- log(address,bool) (internal)
- log(address,bool,address) (internal)
- log(address,bool,address,address) (internal)
- log(address,bool,address,bool) (internal)
- log(address,bool,address,string) (internal)
- log(address,bool,address,uint256) (internal)
- log(address,bool,bool) (internal)
- log(address,bool,bool,address) (internal)
- log(address,bool,bool,bool) (internal)
- log(address,bool,bool,string) (internal)
- log(address,bool,bool,uint256) (internal)
- log(address,bool,string) (internal)
- log(address,bool,string,address) (internal)
- log(address,bool,string,bool) (internal)
- log(address,bool,string,string) (internal)
- log(address,bool,string,uint256) (internal)
- log(address,bool,uint256) (internal)
- log(address,bool,uint256,address) (internal)
- log(address,bool,uint256,bool) (internal)
- log(address,bool,uint256,string) (internal)
- log(address,bool,uint256,uint256) (internal)
- log(address,string) (internal)
- log(address,string,address) (internal)
- log(address,string,address,address) (internal)
- log(address,string,address,bool) (internal)
- log(address,string,address,string) (internal)
- log(address,string,address,uint256) (internal)
- log(address,string,bool) (internal)
- log(address,string,bool,address) (internal)
- log(address,string,bool,bool) (internal)



- `log(address,string,bool,string)` (internal)
- `log(address,string,bool,uint256)` (internal)
- `log(address,string,string)` (internal)
- `log(address,string,string,address)` (internal)
- `log(address,string,string,bool)` (internal)
- `log(address,string,string,string)` (internal)
- `log(address,string,string,uint256)` (internal)
- `log(address,string,uint256)` (internal)
- `log(address,string,uint256,address)` (internal)
- `log(address,string,uint256,bool)` (internal)
- `log(address,string,uint256,string)` (internal)
- `log(address,string,uint256,uint256)` (internal)
- `log(address,uint256)` (internal)
- `log(address,uint256,address)` (internal)
- `log(address,uint256,address,address)` (internal)
- `log(address,uint256,address,bool)` (internal)
- `log(address,uint256,address,string)` (internal)
- `log(address,uint256,address,uint256)` (internal)
- `log(address,uint256,bool)` (internal)
- `log(address,uint256,bool,address)` (internal)
- `log(address,uint256,bool,bool)` (internal)
- `log(address,uint256,bool,string)` (internal)
- `log(address,uint256,bool,uint256)` (internal)
- `log(address,uint256,string)` (internal)
- `log(address,uint256,string,address)` (internal)
- `log(address,uint256,string,bool)` (internal)
- `log(address,uint256,string,string)` (internal)
- `log(address,uint256,string,uint256)` (internal)
- `log(address,uint256,uint256)` (internal)
- `log(address,uint256,uint256,address)` (internal)
- `log(address,uint256,uint256,bool)` (internal)
- `log(address,uint256,uint256,string)` (internal)
- `log(address,uint256,uint256,uint256)` (internal)
- `log(bool)` (internal)
- `log(bool,address)` (internal)
- `log(bool,address,address)` (internal)



- `log(bool,address,address,address)` (internal)
- `log(bool,address,address,bool)` (internal)
- `log(bool,address,address,string)` (internal)
- `log(bool,address,address,uint256)` (internal)
- `log(bool,address,bool)` (internal)
- `log(bool,address,bool,address)` (internal)
- `log(bool,address,bool,bool)` (internal)
- `log(bool,address,bool,string)` (internal)
- `log(bool,address,bool,uint256)` (internal)
- `log(bool,address,string)` (internal)
- `log(bool,address,string,address)` (internal)
- `log(bool,address,string,bool)` (internal)
- `log(bool,address,string,string)` (internal)
- `log(bool,address,string,uint256)` (internal)
- `log(bool,address,uint256)` (internal)
- `log(bool,address,uint256,address)` (internal)
- `log(bool,address,uint256,bool)` (internal)
- `log(bool,address,uint256,string)` (internal)
- `log(bool,address,uint256,uint256)` (internal)
- `log(bool,bool)` (internal)
- `log(bool,bool,address)` (internal)
- `log(bool,bool,address,address)` (internal)
- `log(bool,bool,address,bool)` (internal)
- `log(bool,bool,address,string)` (internal)
- `log(bool,bool,address,uint256)` (internal)
- `log(bool,bool,bool)` (internal)
- `log(bool,bool,bool,address)` (internal)
- `log(bool,bool,bool,bool)` (internal)
- `log(bool,bool,bool,string)` (internal)
- `log(bool,bool,bool,uint256)` (internal)
- `log(bool,bool,string)` (internal)
- `log(bool,bool,string,address)` (internal)
- `log(bool,bool,string,bool)` (internal)
- `log(bool,bool,string,string)` (internal)
- `log(bool,bool,string,uint256)` (internal)
- `log(bool,bool,uint256)` (internal)



- log(bool,bool,uint256,address) (internal)
- log(bool,bool,uint256,bool) (internal)
- log(bool,bool,uint256,string) (internal)
- log(bool,bool,uint256,uint256) (internal)
- log(bool,string) (internal)
- log(bool,string,address) (internal)
- log(bool,string,address,address) (internal)
- log(bool,string,address,bool) (internal)
- log(bool,string,address,string) (internal)
- log(bool,string,address,uint256) (internal)
- log(bool,string,bool) (internal)
- log(bool,string,bool,address) (internal)
- log(bool,string,bool,bool) (internal)
- log(bool,string,bool,string) (internal)
- log(bool,string,bool,uint256) (internal)
- log(bool,string,string) (internal)
- log(bool,string,string,address) (internal)
- log(bool,string,string,bool) (internal)
- log(bool,string,string,string) (internal)
- log(bool,string,string,uint256) (internal)
- log(bool,string,uint256) (internal)
- log(bool,string,uint256,address) (internal)
- log(bool,string,uint256,bool) (internal)
- log(bool,string,uint256,string) (internal)
- log(bool,string,uint256,uint256) (internal)
- log(bool,uint256) (internal)
- log(bool,uint256,address) (internal)
- log(bool,uint256,address,address) (internal)
- log(bool,uint256,address,bool) (internal)
- log(bool,uint256,address,string) (internal)
- log(bool,uint256,address,uint256) (internal)
- log(bool,uint256,bool) (internal)
- log(bool,uint256,bool,address) (internal)
- log(bool,uint256,bool,bool) (internal)
- log(bool,uint256,bool,string) (internal)
- log(bool,uint256,bool,uint256) (internal)



- `log(bool,uint256,string)` (internal)
- `log(bool,uint256,string,address)` (internal)
- `log(bool,uint256,string,bool)` (internal)
- `log(bool,uint256,string,string)` (internal)
- `log(bool,uint256,string,uint256)` (internal)
- `log(bool,uint256,uint256)` (internal)
- `log(bool,uint256,uint256,address)` (internal)
- `log(bool,uint256,uint256,bool)` (internal)
- `log(bool,uint256,uint256,string)` (internal)
- `log(bool,uint256,uint256,uint256)` (internal)
- `log(string)` (internal)
- `log(string,address)` (internal)
- `log(string,address,address)` (internal)
- `log(string,address,address,address)` (internal)
- `log(string,address,address,bool)` (internal)
- `log(string,address,address,string)` (internal)
- `log(string,address,address,uint256)` (internal)
- `log(string,address,bool)` (internal)
- `log(string,address,bool,address)` (internal)
- `log(string,address,bool,bool)` (internal)
- `log(string,address,bool,string)` (internal)
- `log(string,address,bool,uint256)` (internal)
- `log(string,address,string)` (internal)
- `log(string,address,string,address)` (internal)
- `log(string,address,string,bool)` (internal)
- `log(string,address,string,string)` (internal)
- `log(string,address,string,uint256)` (internal)
- `log(string,address,uint256)` (internal)
- `log(string,address,uint256,address)` (internal)
- `log(string,address,uint256,bool)` (internal)
- `log(string,address,uint256,string)` (internal)
- `log(string,address,uint256,uint256)` (internal)
- `log(string,bool)` (internal)
- `log(string,bool,address)` (internal)
- `log(string,bool,address,address)` (internal)
- `log(string,bool,address,bool)` (internal)



- `log(string,bool,address,string)` (internal)
- `log(string,bool,address,uint256)` (internal)
- `log(string,bool,bool)` (internal)
- `log(string,bool,bool,address)` (internal)
- `log(string,bool,bool,bool)` (internal)
- `log(string,bool,bool,string)` (internal)
- `log(string,bool,bool,uint256)` (internal)
- `log(string,bool,string)` (internal)
- `log(string,bool,string,address)` (internal)
- `log(string,bool,string,bool)` (internal)
- `log(string,bool,string,string)` (internal)
- `log(string,bool,string,uint256)` (internal)
- `log(string,bool,uint256)` (internal)
- `log(string,bool,uint256,address)` (internal)
- `log(string,bool,uint256,bool)` (internal)
- `log(string,bool,uint256,string)` (internal)
- `log(string,bool,uint256,uint256)` (internal)
- `log(string,string)` (internal)
- `log(string,string,address)` (internal)
- `log(string,string,address,address)` (internal)
- `log(string,string,address,bool)` (internal)
- `log(string,string,address,string)` (internal)
- `log(string,string,address,uint256)` (internal)
- `log(string,string,bool)` (internal)
- `log(string,string,bool,address)` (internal)
- `log(string,string,bool,bool)` (internal)
- `log(string,string,bool,string)` (internal)
- `log(string,string,bool,uint256)` (internal)
- `log(string,string,string)` (internal)
- `log(string,string,string,address)` (internal)
- `log(string,string,string,bool)` (internal)
- `log(string,string,string,string)` (internal)
- `log(string,string,string,uint256)` (internal)
- `log(string,string,uint256)` (internal)
- `log(string,string,uint256,address)` (internal)
- `log(string,string,uint256,bool)` (internal)



- `log(string,string,uint256,string)` (internal)
- `log(string,string,uint256,uint256)` (internal)
- `log(string,uint256)` (internal)
- `log(string,uint256,address)` (internal)
- `log(string,uint256,address,address)` (internal)
- `log(string,uint256,address,bool)` (internal)
- `log(string,uint256,address,string)` (internal)
- `log(string,uint256,address,uint256)` (internal)
- `log(string,uint256,bool)` (internal)
- `log(string,uint256,bool,address)` (internal)
- `log(string,uint256,bool,bool)` (internal)
- `log(string,uint256,bool,string)` (internal)
- `log(string,uint256,bool,uint256)` (internal)
- `log(string,uint256,string)` (internal)
- `log(string,uint256,string,address)` (internal)
- `log(string,uint256,string,bool)` (internal)
- `log(string,uint256,string,string)` (internal)
- `log(string,uint256,string,uint256)` (internal)
- `log(string,uint256,uint256)` (internal)
- `log(string,uint256,uint256,address)` (internal)
- `log(string,uint256,uint256,bool)` (internal)
- `log(string,uint256,uint256,string)` (internal)
- `log(string,uint256,uint256,uint256)` (internal)
- `log(uint256)` (internal)
- `log(uint256,address)` (internal)
- `log(uint256,address,address)` (internal)
- `log(uint256,address,address,address)` (internal)
- `log(uint256,address,address,bool)` (internal)
- `log(uint256,address,address,string)` (internal)
- `log(uint256,address,address,uint256)` (internal)
- `log(uint256,address,bool)` (internal)
- `log(uint256,address,bool,address)` (internal)
- `log(uint256,address,bool,bool)` (internal)
- `log(uint256,address,bool,string)` (internal)
- `log(uint256,address,bool,uint256)` (internal)
- `log(uint256,address,string)` (internal)



- `log(uint256,address,string,address)` (internal)
- `log(uint256,address,string,bool)` (internal)
- `log(uint256,address,string,string)` (internal)
- `log(uint256,address,string,uint256)` (internal)
- `log(uint256,address,uint256)` (internal)
- `log(uint256,address,uint256,address)` (internal)
- `log(uint256,address,uint256,bool)` (internal)
- `log(uint256,address,uint256,string)` (internal)
- `log(uint256,address,uint256,uint256)` (internal)
- `log(uint256,bool)` (internal)
- `log(uint256,bool,address)` (internal)
- `log(uint256,bool,address,address)` (internal)
- `log(uint256,bool,address,bool)` (internal)
- `log(uint256,bool,address,string)` (internal)
- `log(uint256,bool,address,uint256)` (internal)
- `log(uint256,bool,bool)` (internal)
- `log(uint256,bool,bool,address)` (internal)
- `log(uint256,bool,bool,bool)` (internal)
- `log(uint256,bool,bool,string)` (internal)
- `log(uint256,bool,bool,uint256)` (internal)
- `log(uint256,bool,string)` (internal)
- `log(uint256,bool,string,address)` (internal)
- `log(uint256,bool,string,bool)` (internal)
- `log(uint256,bool,string,string)` (internal)
- `log(uint256,bool,string,uint256)` (internal)
- `log(uint256,bool,uint256)` (internal)
- `log(uint256,bool,uint256,address)` (internal)
- `log(uint256,bool,uint256,bool)` (internal)
- `log(uint256,bool,uint256,string)` (internal)
- `log(uint256,bool,uint256,uint256)` (internal)
- `log(uint256,string)` (internal)
- `log(uint256,string,address)` (internal)
- `log(uint256,string,address,address)` (internal)
- `log(uint256,string,address,bool)` (internal)
- `log(uint256,string,address,string)` (internal)
- `log(uint256,string,address,uint256)` (internal)



- `log(uint256,string,bool)` (internal)
- `log(uint256,string,bool,address)` (internal)
- `log(uint256,string,bool,bool)` (internal)
- `log(uint256,string,bool,string)` (internal)
- `log(uint256,string,bool,uint256)` (internal)
- `log(uint256,string,string)` (internal)
- `log(uint256,string,string,address)` (internal)
- `log(uint256,string,string,bool)` (internal)
- `log(uint256,string,string,string)` (internal)
- `log(uint256,string,string,uint256)` (internal)
- `log(uint256,string,uint256)` (internal)
- `log(uint256,string,uint256,address)` (internal)
- `log(uint256,string,uint256,bool)` (internal)
- `log(uint256,string,uint256,string)` (internal)
- `log(uint256,string,uint256,uint256)` (internal)
- `log(uint256,uint256)` (internal)
- `log(uint256,uint256,address)` (internal)
- `log(uint256,uint256,address,address)` (internal)
- `log(uint256,uint256,address,bool)` (internal)
- `log(uint256,uint256,address,string)` (internal)
- `log(uint256,uint256,address,uint256)` (internal)
- `log(uint256,uint256,bool)` (internal)
- `log(uint256,uint256,bool,address)` (internal)
- `log(uint256,uint256,bool,bool)` (internal)
- `log(uint256,uint256,bool,string)` (internal)
- `log(uint256,uint256,bool,uint256)` (internal)
- `log(uint256,uint256,string)` (internal)
- `log(uint256,uint256,string,address)` (internal)
- `log(uint256,uint256,string,bool)` (internal)
- `log(uint256,uint256,string,string)` (internal)
- `log(uint256,uint256,string,uint256)` (internal)
- `log(uint256,uint256,uint256)` (internal)
- `log(uint256,uint256,uint256,address)` (internal)
- `log(uint256,uint256,uint256,bool)` (internal)
- `log(uint256,uint256,uint256,string)` (internal)
- `log(uint256,uint256,uint256,uint256)` (internal)



- `logAddress(address)` (internal)
- `logBool(bool)` (internal)
- `logBytes(bytes)` (internal)
- `logBytes1(bytes1)` (internal)
- `logBytes10(bytes10)` (internal)
- `logBytes11(bytes11)` (internal)
- `logBytes12(bytes12)` (internal)
- `logBytes13(bytes13)` (internal)
- `logBytes14(bytes14)` (internal)
- `logBytes15(bytes15)` (internal)
- `logBytes16(bytes16)` (internal)
- `logBytes17(bytes17)` (internal)
- `logBytes18(bytes18)` (internal)
- `logBytes19(bytes19)` (internal)
- `logBytes2(bytes2)` (internal)
- `logBytes20(bytes20)` (internal)
- `logBytes21(bytes21)` (internal)
- `logBytes22(bytes22)` (internal)
- `logBytes23(bytes23)` (internal)
- `logBytes24(bytes24)` (internal)
- `logBytes25(bytes25)` (internal)
- `logBytes26(bytes26)` (internal)
- `logBytes27(bytes27)` (internal)
- `logBytes28(bytes28)` (internal)
- `logBytes29(bytes29)` (internal)
- `logBytes3(bytes3)` (internal)
- `logBytes30(bytes30)` (internal)
- `logBytes31(bytes31)` (internal)
- `logBytes32(bytes32)` (internal)
- `logBytes4(bytes4)` (internal)
- `logBytes5(bytes5)` (internal)
- `logBytes6(bytes6)` (internal)
- `logBytes7(bytes7)` (internal)
- `logBytes8(bytes8)` (internal)
- `logBytes9(bytes9)` (internal)
- `logInt(int256)` (internal)



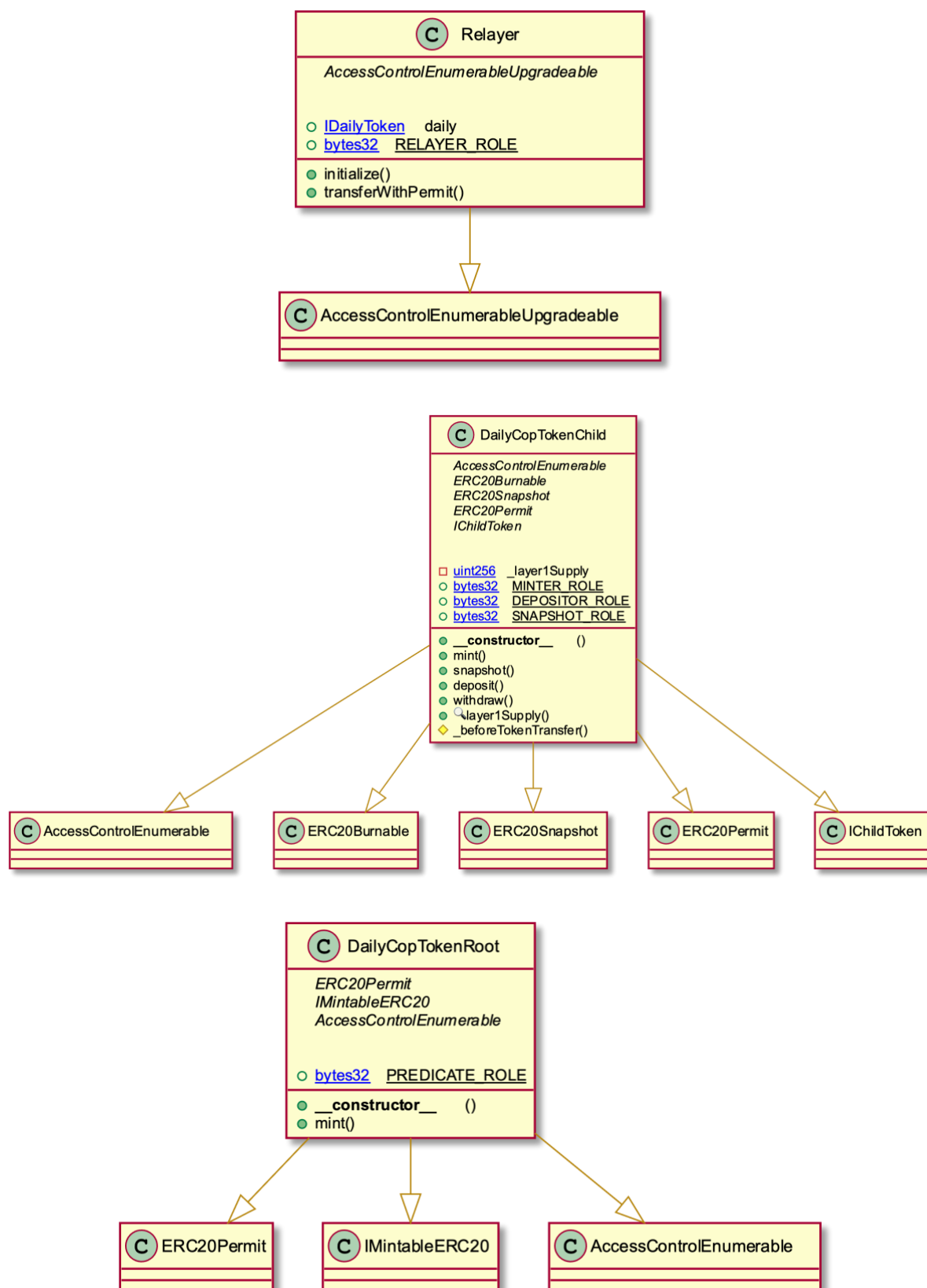
- logString(string) (internal)
- logUint(uint256) (internal)
- Contract DailyCopTokenChild (Most derived contract)
 - From ERC20Permit
 - DOMAIN_SEPARATOR() (external)
 - _useNonce(address) (internal)
 - constructor(string) (internal)
 - nonces(address) (public)
 - permit(address,address,uint256,uint256,uint8,bytes32,bytes32) (public)
 - From EIP712
 - _buildDomainSeparator(bytes32,bytes32,bytes32) (private)
 - _domainSeparatorV4() (internal)
 - _hashTypedDataV4(bytes32) (internal)
 - constructor(string,string) (internal)
 - From ERC20
 - _afterTokenTransfer(address,address,uint256) (internal)
 - _approve(address,address,uint256) (internal)
 - _burn(address,uint256) (internal)
 - _mint(address,uint256) (internal)
 - _transfer(address,address,uint256) (internal)
 - allowance(address,address) (public)
 - approve(address,uint256) (public)
 - balanceOf(address) (public)
 - decimals() (public)
 - decreaseAllowance(address,uint256) (public)
 - increaseAllowance(address,uint256) (public)
 - name() (public)
 - symbol() (public)
 - totalSupply() (public)
 - transfer(address,uint256) (public)
 - transferFrom(address,address,uint256) (public)
 - From Context
 - _msgData() (internal)
 - _msgSender() (internal)
 - From ERC20Snapshot



- `_getCurrentSnapshotId()` (internal)
- `_lastSnapshotId(uint256[])` (private)
- `_snapshot()` (internal)
- `_updateAccountSnapshot(address)` (private)
- `_updateSnapshot(ERC20Snapshot.Snapshots,uint256)` (private)
- `_updateTotalSupplySnapshot()` (private)
- `_valueAt(uint256,ERC20Snapshot.Snapshots)` (private)
- `balanceOfAt(address,uint256)` (public)
- `totalSupplyAt(uint256)` (public)
- From `ERC20Burnable`
 - `burn(uint256)` (public)
 - `burnFrom(address,uint256)` (public)
- From `AccessControlEnumerable`
 - `_setupRole(bytes32,address)` (internal)
 - `getRoleMember(bytes32,uint256)` (public)
 - `getRoleMemberCount(bytes32)` (public)
 - `grantRole(bytes32,address)` (public)
 - `renounceRole(bytes32,address)` (public)
 - `revokeRole(bytes32,address)` (public)
 - `supportsInterface(bytes4)` (public)
- From `AccessControl`
 - `_checkRole(bytes32,address)` (internal)
 - `_grantRole(bytes32,address)` (private)
 - `_revokeRole(bytes32,address)` (private)
 - `_setRoleAdmin(bytes32,bytes32)` (internal)
 - `getRoleAdmin(bytes32)` (public)
 - `hasRole(bytes32,address)` (public)
- From `DailyCopTokenChild`
 - `_beforeTokenTransfer(address,address,uint256)` (internal)
 - `constructor(address,address)` (public)
 - `deposit(address,bytes)` (external)
 - `layer1Supply()` (public)
 - `mint(address,uint256)` (public)
 - `snapshot()` (public)
 - `withdraw(uint256)` (external)



Inheritance Graph and UML Diagram



Slither Results

Relayer

```
> slither contracts/Relayer-flattened.sol
```

Compilation warnings/errors on contracts/Relayer-flattened.sol:

Warning: SPDX license identifier not provided in source file. Before publishing, consider adding a comment containing "SPDX-License-Identifier: <SPDX-License>" to each source file. Use "SPDX-License-Identifier: UNLICENSED" for non-open-source code. Please see <https://spdx.org> for more information.

```
--> contracts/Relayer-flattened.sol
```

INFO:Detectors:

AccessControlUpgradeable.__gap (contracts/Relayer-flattened.sol#943) shadows:

- ERC165Upgradeable.__gap (contracts/Relayer-flattened.sol#268)
- ContextUpgradeable.__gap (contracts/Relayer-flattened.sol#377)

AccessControlEnumerableUpgradeable.__gap (contracts/Relayer-flattened.sol#1062) shadows:

- AccessControlUpgradeable.__gap (contracts/Relayer-flattened.sol#943)
- ERC165Upgradeable.__gap (contracts/Relayer-flattened.sol#268)
- ContextUpgradeable.__gap (contracts/Relayer-flattened.sol#377)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#state-variable-shadowing>

INFO:Detectors:

Relayer.transferWithPermit(address,address,uint256,uint256,uint8,bytes32,bytes32) (contracts/Relayer-flattened.sol#2633-2644) ignores return value by daily.transferFrom(owner,recipient,value) (contracts/Relayer-flattened.sol#2643)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#unchecked-transfer>

INFO:Detectors:

AccessControlEnumerableUpgradeable.grantRole(bytes32,address) (contracts/Relayer-flattened.sol#1034-1037) ignores return value by _roleMembers[role].add(account) (contracts/Relayer-flattened.sol#1036)

AccessControlEnumerableUpgradeable.revokeRole(bytes32,address) (contracts/Relayer-flattened.sol#1042-1045) ignores return value by _roleMembers[role].remove(account) (contracts/Relayer-flattened.sol#1044)

AccessControlEnumerableUpgradeable.renounceRole(bytes32,address) (contracts/Relayer-flattened.sol#1050-1053) ignores return value by _roleMembers[role].remove(account) (contracts/Relayer-flattened.sol#1052)

AccessControlEnumerableUpgradeable._setupRole(bytes32,address) (contracts/Relayer-flattened.sol#1058-1061) ignores return value by _roleMembers[role].add(account) (contracts/Relayer-flattened.sol#1060)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#unused-return>

INFO:Detectors:

console._sendLogPayload(bytes) (contracts/Relayer-flattened.sol#1076-1083) uses assembly

- INLINE ASM (contracts/Relayer-flattened.sol#1079-1082)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-usage>

INFO:Detectors:

Different versions of Solidity is used:

- Version used: ['0.8.4', '>=0.4.22<0.9.0', '^0.8.0']
- ^0.8.0 (contracts/Relayer-flattened.sol#3)



- ^0.8.0 (contracts/Relayer-flattened.sol#68)
- ^0.8.0 (contracts/Relayer-flattened.sol#154)
- ^0.8.0 (contracts/Relayer-flattened.sol#206)
- ^0.8.0 (contracts/Relayer-flattened.sol#236)
- ^0.8.0 (contracts/Relayer-flattened.sol#278)
- ^0.8.0 (contracts/Relayer-flattened.sol#350)
- ^0.8.0 (contracts/Relayer-flattened.sol#387)
- ^0.8.0 (contracts/Relayer-flattened.sol#687)
- 0.8.4 (contracts/Relayer-flattened.sol#952)
- ^0.8.0 (contracts/Relayer-flattened.sol#968)
- >=0.4.22<0.9.0 (contracts/Relayer-flattened.sol#1071)
- 0.8.4 (contracts/Relayer-flattened.sol#2607)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#different-pragma-directives-are-used>

INFO:Detectors:

AccessControlUpgradeable.__AccessControl_init() (contracts/Relayer-flattened.sol#748-752) is never used and should be removed

AccessControlUpgradeable._setRoleAdmin(bytes32,bytes32) (contracts/Relayer-flattened.sol#925-928) is never used and should be removed

ContextUpgradeable.__Context_init() (contracts/Relayer-flattened.sol#364-366) is never used and should be removed

ContextUpgradeable._msgData() (contracts/Relayer-flattened.sol#374-376) is never used and should be removed

ERC165Upgradeable.__ERC165_init() (contracts/Relayer-flattened.sol#256-258) is never used and should be removed

EnumerableSetUpgradeable.add(EnumerableSetUpgradeable.Bytes32Set,bytes32) (contracts/Relayer-flattened.sol#529-531) is never used and should be removed

EnumerableSetUpgradeable.add(EnumerableSetUpgradeable.UintSet,uint256) (contracts/Relayer-flattened.sol#637-639) is never used and should be removed

EnumerableSetUpgradeable.at(EnumerableSetUpgradeable.Bytes32Set,uint256) (contracts/Relayer-flattened.sol#567-569) is never used and should be removed

EnumerableSetUpgradeable.at(EnumerableSetUpgradeable.UintSet,uint256) (contracts/Relayer-flattened.sol#675-677) is never used and should be removed

EnumerableSetUpgradeable.contains(EnumerableSetUpgradeable.AddressSet,address) (contracts/Relayer-flattened.sol#600-602) is never used and should be removed

EnumerableSetUpgradeable.contains(EnumerableSetUpgradeable.Bytes32Set,bytes32) (contracts/Relayer-flattened.sol#546-548) is never used and should be removed

EnumerableSetUpgradeable.contains(EnumerableSetUpgradeable.UintSet,uint256) (contracts/Relayer-flattened.sol#654-656) is never used and should be removed

EnumerableSetUpgradeable.length(EnumerableSetUpgradeable.Bytes32Set) (contracts/Relayer-flattened.sol#553-555) is never used and should be removed

EnumerableSetUpgradeable.length(EnumerableSetUpgradeable.UintSet) (contracts/Relayer-flattened.sol#661-663) is never used and should be removed

EnumerableSetUpgradeable.remove(EnumerableSetUpgradeable.Bytes32Set,bytes32) (contracts/Relayer-flattened.sol#539-541) is never used and should be removed

EnumerableSetUpgradeable.remove(EnumerableSetUpgradeable.UintSet,uint256) (contracts/Relayer-flattened.sol#647-649) is never used and should be removed

StringsUpgradeable.toHexString(uint256) (contracts/Relayer-flattened.sol#314-325) is never used and should be removed

StringsUpgradeable.toString(uint256) (contracts/Relayer-flattened.sol#289-309) is never used and should be removed

console._sendLogPayload(bytes) (contracts/Relayer-flattened.sol#1076-1083) is never used and should be removed

console.log() (contracts/Relayer-flattened.sol#1085-1087) is never used and should be removed

console.log(address) (contracts/Relayer-flattened.sol#1253-1255) is never used and should be removed

console.log(address,address) (contracts/Relayer-flattened.sol#1317-1319) is never used and should be removed



`console.log(address,address,address)` (contracts/Relayer-flattened.sol#1573-1575) is never used and should be removed
`console.log(address,address,address,address)` (contracts/Relayer-flattened.sol#2597-2599) is never used and should be removed
`console.log(address,address,address,bool)` (contracts/Relayer-flattened.sol#2593-2595) is never used and should be removed
`console.log(address,address,address,string)` (contracts/Relayer-flattened.sol#2589-2591) is never used and should be removed
`console.log(address,address,address,uint256)` (contracts/Relayer-flattened.sol#2585-2587) is never used and should be removed
`console.log(address,address,bool)` (contracts/Relayer-flattened.sol#1569-1571) is never used and should be removed
`console.log(address,address,bool,address)` (contracts/Relayer-flattened.sol#2581-2583) is never used and should be removed
`console.log(address,address,bool,bool)` (contracts/Relayer-flattened.sol#2577-2579) is never used and should be removed
`console.log(address,address,bool,string)` (contracts/Relayer-flattened.sol#2573-2575) is never used and should be removed
`console.log(address,address,bool,uint256)` (contracts/Relayer-flattened.sol#2569-2571) is never used and should be removed
`console.log(address,address,string)` (contracts/Relayer-flattened.sol#1565-1567) is never used and should be removed
`console.log(address,address,string,address)` (contracts/Relayer-flattened.sol#2565-2567) is never used and should be removed
`console.log(address,address,string,bool)` (contracts/Relayer-flattened.sol#2561-2563) is never used and should be removed
`console.log(address,address,string,string)` (contracts/Relayer-flattened.sol#2557-2559) is never used and should be removed
`console.log(address,address,string,uint256)` (contracts/Relayer-flattened.sol#2553-2555) is never used and should be removed
`console.log(address,address,uint256)` (contracts/Relayer-flattened.sol#1561-1563) is never used and should be removed
`console.log(address,address,uint256,address)` (contracts/Relayer-flattened.sol#2549-2551) is never used and should be removed
`console.log(address,address,uint256,bool)` (contracts/Relayer-flattened.sol#2545-2547) is never used and should be removed
`console.log(address,address,uint256,string)` (contracts/Relayer-flattened.sol#2541-2543) is never used and should be removed
`console.log(address,address,uint256,uint256)` (contracts/Relayer-flattened.sol#2537-2539) is never used and should be removed
`console.log(address,bool)` (contracts/Relayer-flattened.sol#1313-1315) is never used and should be removed
`console.log(address,bool,address)` (contracts/Relayer-flattened.sol#1557-1559) is never used and should be removed
`console.log(address,bool,address,address)` (contracts/Relayer-flattened.sol#2533-2535) is never used and should be removed
`console.log(address,bool,address,bool)` (contracts/Relayer-flattened.sol#2529-2531) is never used and should be removed
`console.log(address,bool,address,string)` (contracts/Relayer-flattened.sol#2525-2527) is never used and should be removed
`console.log(address,bool,address,uint256)` (contracts/Relayer-flattened.sol#2521-2523) is never used and should be removed
`console.log(address,bool,bool)` (contracts/Relayer-flattened.sol#1553-1555) is never used and should be removed
`console.log(address,bool,bool,address)` (contracts/Relayer-flattened.sol#2517-2519) is never used and should be removed
`console.log(address,bool,bool,bool)` (contracts/Relayer-flattened.sol#2513-2515) is never used and should be removed



console.log(address,bool,bool,string) (contracts/Relayer-flattened.sol#2509-2511) is never used and should be removed
console.log(address,bool,bool,uint256) (contracts/Relayer-flattened.sol#2505-2507) is never used and should be removed
console.log(address,bool,string) (contracts/Relayer-flattened.sol#1549-1551) is never used and should be removed
console.log(address,bool,string,address) (contracts/Relayer-flattened.sol#2501-2503) is never used and should be removed
console.log(address,bool,string,bool) (contracts/Relayer-flattened.sol#2497-2499) is never used and should be removed
console.log(address,bool,string,string) (contracts/Relayer-flattened.sol#2493-2495) is never used and should be removed
console.log(address,bool,string,uint256) (contracts/Relayer-flattened.sol#2489-2491) is never used and should be removed
console.log(address,bool,uint256) (contracts/Relayer-flattened.sol#1545-1547) is never used and should be removed
console.log(address,bool,uint256,address) (contracts/Relayer-flattened.sol#2485-2487) is never used and should be removed
console.log(address,bool,uint256,bool) (contracts/Relayer-flattened.sol#2481-2483) is never used and should be removed
console.log(address,bool,uint256,string) (contracts/Relayer-flattened.sol#2477-2479) is never used and should be removed
console.log(address,bool,uint256,uint256) (contracts/Relayer-flattened.sol#2473-2475) is never used and should be removed
console.log(address,string) (contracts/Relayer-flattened.sol#1309-1311) is never used and should be removed
console.log(address,string,address) (contracts/Relayer-flattened.sol#1541-1543) is never used and should be removed
console.log(address,string,address,address) (contracts/Relayer-flattened.sol#2469-2471) is never used and should be removed
console.log(address,string,address,bool) (contracts/Relayer-flattened.sol#2465-2467) is never used and should be removed
console.log(address,string,address,string) (contracts/Relayer-flattened.sol#2461-2463) is never used and should be removed
console.log(address,string,address,uint256) (contracts/Relayer-flattened.sol#2457-2459) is never used and should be removed
console.log(address,string,bool) (contracts/Relayer-flattened.sol#1537-1539) is never used and should be removed
console.log(address,string,bool,address) (contracts/Relayer-flattened.sol#2453-2455) is never used and should be removed
console.log(address,string,bool,bool) (contracts/Relayer-flattened.sol#2449-2451) is never used and should be removed
console.log(address,string,bool,string) (contracts/Relayer-flattened.sol#2445-2447) is never used and should be removed
console.log(address,string,bool,uint256) (contracts/Relayer-flattened.sol#2441-2443) is never used and should be removed
console.log(address,string,string) (contracts/Relayer-flattened.sol#1533-1535) is never used and should be removed
console.log(address,string,string,address) (contracts/Relayer-flattened.sol#2437-2439) is never used and should be removed
console.log(address,string,string,bool) (contracts/Relayer-flattened.sol#2433-2435) is never used and should be removed
console.log(address,string,string,string) (contracts/Relayer-flattened.sol#2429-2431) is never used and should be removed
console.log(address,string,string,uint256) (contracts/Relayer-flattened.sol#2425-2427) is never used and should be removed
console.log(address,string,uint256) (contracts/Relayer-flattened.sol#1529-1531) is never used and should be removed



console.log(address,string,uint256,address) (contracts/Relayer-flattened.sol#2421-2423) is never used and should be removed
console.log(address,string,uint256,bool) (contracts/Relayer-flattened.sol#2417-2419) is never used and should be removed
console.log(address,string,uint256,string) (contracts/Relayer-flattened.sol#2413-2415) is never used and should be removed
console.log(address,string,uint256,uint256) (contracts/Relayer-flattened.sol#2409-2411) is never used and should be removed
console.log(address,uint256) (contracts/Relayer-flattened.sol#1305-1307) is never used and should be removed
console.log(address,uint256,address) (contracts/Relayer-flattened.sol#1525-1527) is never used and should be removed
console.log(address,uint256,address,address) (contracts/Relayer-flattened.sol#2405-2407) is never used and should be removed
console.log(address,uint256,address,bool) (contracts/Relayer-flattened.sol#2401-2403) is never used and should be removed
console.log(address,uint256,address,string) (contracts/Relayer-flattened.sol#2397-2399) is never used and should be removed
console.log(address,uint256,address,uint256) (contracts/Relayer-flattened.sol#2393-2395) is never used and should be removed
console.log(address,uint256,bool) (contracts/Relayer-flattened.sol#1521-1523) is never used and should be removed
console.log(address,uint256,bool,address) (contracts/Relayer-flattened.sol#2389-2391) is never used and should be removed
console.log(address,uint256,bool,bool) (contracts/Relayer-flattened.sol#2385-2387) is never used and should be removed
console.log(address,uint256,bool,string) (contracts/Relayer-flattened.sol#2381-2383) is never used and should be removed
console.log(address,uint256,bool,uint256) (contracts/Relayer-flattened.sol#2377-2379) is never used and should be removed
console.log(address,uint256,string) (contracts/Relayer-flattened.sol#1517-1519) is never used and should be removed
console.log(address,uint256,string,address) (contracts/Relayer-flattened.sol#2373-2375) is never used and should be removed
console.log(address,uint256,string,bool) (contracts/Relayer-flattened.sol#2369-2371) is never used and should be removed
console.log(address,uint256,string,string) (contracts/Relayer-flattened.sol#2365-2367) is never used and should be removed
console.log(address,uint256,string,uint256) (contracts/Relayer-flattened.sol#2361-2363) is never used and should be removed
console.log(address,uint256,uint256) (contracts/Relayer-flattened.sol#1513-1515) is never used and should be removed
console.log(address,uint256,uint256,address) (contracts/Relayer-flattened.sol#2357-2359) is never used and should be removed
console.log(address,uint256,uint256,bool) (contracts/Relayer-flattened.sol#2353-2355) is never used and should be removed
console.log(address,uint256,uint256,string) (contracts/Relayer-flattened.sol#2349-2351) is never used and should be removed
console.log(address,uint256,uint256,uint256) (contracts/Relayer-flattened.sol#2345-2347) is never used and should be removed
console.log(bool) (contracts/Relayer-flattened.sol#1249-1251) is never used and should be removed
console.log(bool,address) (contracts/Relayer-flattened.sol#1301-1303) is never used and should be removed
console.log(bool,address,address) (contracts/Relayer-flattened.sol#1509-1511) is never used and should be removed
console.log(bool,address,address,address) (contracts/Relayer-flattened.sol#2341-2343) is never used and should be removed



```

console.log(bool,address,address,bool) (contracts/Relayer-flattened.sol#2337-2339)
is never used and should be removed
console.log(bool,address,address,string) (contracts/Relayer-flattened.sol#2333-
2335) is never used and should be removed
console.log(bool,address,address,uint256) (contracts/Relayer-flattened.sol#2329-
2331) is never used and should be removed
console.log(bool,address,bool) (contracts/Relayer-flattened.sol#1505-1507) is
never used and should be removed
console.log(bool,address,bool,address) (contracts/Relayer-flattened.sol#2325-2327)
is never used and should be removed
console.log(bool,address,bool,bool) (contracts/Relayer-flattened.sol#2321-2323) is
never used and should be removed
console.log(bool,address,bool,string) (contracts/Relayer-flattened.sol#2317-2319)
is never used and should be removed
console.log(bool,address,bool,uint256) (contracts/Relayer-flattened.sol#2313-2315)
is never used and should be removed
console.log(bool,address,string) (contracts/Relayer-flattened.sol#1501-1503) is
never used and should be removed
console.log(bool,address,string,address) (contracts/Relayer-flattened.sol#2309-
2311) is never used and should be removed
console.log(bool,address,string,bool) (contracts/Relayer-flattened.sol#2305-2307)
is never used and should be removed
console.log(bool,address,string,string) (contracts/Relayer-flattened.sol#2301-
2303) is never used and should be removed
console.log(bool,address,string,uint256) (contracts/Relayer-flattened.sol#2297-
2299) is never used and should be removed
console.log(bool,address,uint256) (contracts/Relayer-flattened.sol#1497-1499) is
never used and should be removed
console.log(bool,address,uint256,address) (contracts/Relayer-flattened.sol#2293-
2295) is never used and should be removed
console.log(bool,address,uint256,bool) (contracts/Relayer-flattened.sol#2289-2291)
is never used and should be removed
console.log(bool,address,uint256,string) (contracts/Relayer-flattened.sol#2285-
2287) is never used and should be removed
console.log(bool,address,uint256,uint256) (contracts/Relayer-flattened.sol#2281-
2283) is never used and should be removed
console.log(bool,bool) (contracts/Relayer-flattened.sol#1297-1299) is never used
and should be removed
console.log(bool,bool,address) (contracts/Relayer-flattened.sol#1493-1495) is
never used and should be removed
console.log(bool,bool,address,address) (contracts/Relayer-flattened.sol#2277-2279)
is never used and should be removed
console.log(bool,bool,address,bool) (contracts/Relayer-flattened.sol#2273-2275) is
never used and should be removed
console.log(bool,bool,address,string) (contracts/Relayer-flattened.sol#2269-2271)
is never used and should be removed
console.log(bool,bool,address,uint256) (contracts/Relayer-flattened.sol#2265-2267)
is never used and should be removed
console.log(bool,bool,bool) (contracts/Relayer-flattened.sol#1489-1491) is never
used and should be removed
console.log(bool,bool,bool,address) (contracts/Relayer-flattened.sol#2261-2263) is
never used and should be removed
console.log(bool,bool,bool,bool) (contracts/Relayer-flattened.sol#2257-2259) is
never used and should be removed
console.log(bool,bool,bool,string) (contracts/Relayer-flattened.sol#2253-2255) is
never used and should be removed
console.log(bool,bool,bool,uint256) (contracts/Relayer-flattened.sol#2249-2251) is
never used and should be removed

```



console.log(bool,bool,string) (contracts/Relayer-flattened.sol#1485-1487) is never used and should be removed

console.log(bool,bool,string,address) (contracts/Relayer-flattened.sol#2245-2247) is never used and should be removed

console.log(bool,bool,string,bool) (contracts/Relayer-flattened.sol#2241-2243) is never used and should be removed

console.log(bool,bool,string,string) (contracts/Relayer-flattened.sol#2237-2239) is never used and should be removed

console.log(bool,bool,string,uint256) (contracts/Relayer-flattened.sol#2233-2235) is never used and should be removed

console.log(bool,bool,uint256) (contracts/Relayer-flattened.sol#1481-1483) is never used and should be removed

console.log(bool,bool,uint256,address) (contracts/Relayer-flattened.sol#2229-2231) is never used and should be removed

console.log(bool,bool,uint256,bool) (contracts/Relayer-flattened.sol#2225-2227) is never used and should be removed

console.log(bool,bool,uint256,string) (contracts/Relayer-flattened.sol#2221-2223) is never used and should be removed

console.log(bool,bool,uint256,uint256) (contracts/Relayer-flattened.sol#2217-2219) is never used and should be removed

console.log(bool,string) (contracts/Relayer-flattened.sol#1293-1295) is never used and should be removed

console.log(bool,string,address) (contracts/Relayer-flattened.sol#1477-1479) is never used and should be removed

console.log(bool,string,address,address) (contracts/Relayer-flattened.sol#2213-2215) is never used and should be removed

console.log(bool,string,address,bool) (contracts/Relayer-flattened.sol#2209-2211) is never used and should be removed

console.log(bool,string,address,string) (contracts/Relayer-flattened.sol#2205-2207) is never used and should be removed

console.log(bool,string,address,uint256) (contracts/Relayer-flattened.sol#2201-2203) is never used and should be removed

console.log(bool,string,bool) (contracts/Relayer-flattened.sol#1473-1475) is never used and should be removed

console.log(bool,string,bool,address) (contracts/Relayer-flattened.sol#2197-2199) is never used and should be removed

console.log(bool,string,bool,bool) (contracts/Relayer-flattened.sol#2193-2195) is never used and should be removed

console.log(bool,string,bool,string) (contracts/Relayer-flattened.sol#2189-2191) is never used and should be removed

console.log(bool,string,bool,uint256) (contracts/Relayer-flattened.sol#2185-2187) is never used and should be removed

console.log(bool,string,string) (contracts/Relayer-flattened.sol#1469-1471) is never used and should be removed

console.log(bool,string,string,address) (contracts/Relayer-flattened.sol#2181-2183) is never used and should be removed

console.log(bool,string,string,bool) (contracts/Relayer-flattened.sol#2177-2179) is never used and should be removed

console.log(bool,string,string,string) (contracts/Relayer-flattened.sol#2173-2175) is never used and should be removed

console.log(bool,string,string,uint256) (contracts/Relayer-flattened.sol#2169-2171) is never used and should be removed

console.log(bool,string,uint256) (contracts/Relayer-flattened.sol#1465-1467) is never used and should be removed

console.log(bool,string,uint256,address) (contracts/Relayer-flattened.sol#2165-2167) is never used and should be removed

console.log(bool,string,uint256,bool) (contracts/Relayer-flattened.sol#2161-2163) is never used and should be removed




```

console.log(bool,string,uint256,string) (contracts/Relayer-flattened.sol#2157-2159) is never used and should be removed
console.log(bool,string,uint256,uint256) (contracts/Relayer-flattened.sol#2153-2155) is never used and should be removed
console.log(bool,uint256) (contracts/Relayer-flattened.sol#1289-1291) is never used and should be removed
console.log(bool,uint256,address) (contracts/Relayer-flattened.sol#1461-1463) is never used and should be removed
console.log(bool,uint256,address,address) (contracts/Relayer-flattened.sol#2149-2151) is never used and should be removed
console.log(bool,uint256,address,bool) (contracts/Relayer-flattened.sol#2145-2147) is never used and should be removed
console.log(bool,uint256,address,string) (contracts/Relayer-flattened.sol#2141-2143) is never used and should be removed
console.log(bool,uint256,address,uint256) (contracts/Relayer-flattened.sol#2137-2139) is never used and should be removed
console.log(bool,uint256,bool) (contracts/Relayer-flattened.sol#1457-1459) is never used and should be removed
console.log(bool,uint256,bool,address) (contracts/Relayer-flattened.sol#2133-2135) is never used and should be removed
console.log(bool,uint256,bool,bool) (contracts/Relayer-flattened.sol#2129-2131) is never used and should be removed
console.log(bool,uint256,bool,string) (contracts/Relayer-flattened.sol#2125-2127) is never used and should be removed
console.log(bool,uint256,bool,uint256) (contracts/Relayer-flattened.sol#2121-2123) is never used and should be removed
console.log(bool,uint256,string) (contracts/Relayer-flattened.sol#1453-1455) is never used and should be removed
console.log(bool,uint256,string,address) (contracts/Relayer-flattened.sol#2117-2119) is never used and should be removed
console.log(bool,uint256,string,bool) (contracts/Relayer-flattened.sol#2113-2115) is never used and should be removed
console.log(bool,uint256,string,string) (contracts/Relayer-flattened.sol#2109-2111) is never used and should be removed
console.log(bool,uint256,string,uint256) (contracts/Relayer-flattened.sol#2105-2107) is never used and should be removed
console.log(bool,uint256,uint256) (contracts/Relayer-flattened.sol#1449-1451) is never used and should be removed
console.log(bool,uint256,uint256,address) (contracts/Relayer-flattened.sol#2101-2103) is never used and should be removed
console.log(bool,uint256,uint256,bool) (contracts/Relayer-flattened.sol#2097-2099) is never used and should be removed
console.log(bool,uint256,uint256,string) (contracts/Relayer-flattened.sol#2093-2095) is never used and should be removed
console.log(bool,uint256,uint256,uint256) (contracts/Relayer-flattened.sol#2089-2091) is never used and should be removed
console.log(string) (contracts/Relayer-flattened.sol#1245-1247) is never used and should be removed
console.log(string,address) (contracts/Relayer-flattened.sol#1285-1287) is never used and should be removed
console.log(string,address,address) (contracts/Relayer-flattened.sol#1445-1447) is never used and should be removed
console.log(string,address,address,address) (contracts/Relayer-flattened.sol#2085-2087) is never used and should be removed
console.log(string,address,address,bool) (contracts/Relayer-flattened.sol#2081-2083) is never used and should be removed
console.log(string,address,address,string) (contracts/Relayer-flattened.sol#2077-2079) is never used and should be removed

```



```

console.log(string,address,address,uint256) (contracts/Relayer-flattened.sol#2073-2075) is never used and should be removed
console.log(string,address,bool) (contracts/Relayer-flattened.sol#1441-1443) is never used and should be removed
console.log(string,address,bool,address) (contracts/Relayer-flattened.sol#2069-2071) is never used and should be removed
console.log(string,address,bool,bool) (contracts/Relayer-flattened.sol#2065-2067) is never used and should be removed
console.log(string,address,bool,string) (contracts/Relayer-flattened.sol#2061-2063) is never used and should be removed
console.log(string,address,bool,uint256) (contracts/Relayer-flattened.sol#2057-2059) is never used and should be removed
console.log(string,address,string) (contracts/Relayer-flattened.sol#1437-1439) is never used and should be removed
console.log(string,address,string,address) (contracts/Relayer-flattened.sol#2053-2055) is never used and should be removed
console.log(string,address,string,bool) (contracts/Relayer-flattened.sol#2049-2051) is never used and should be removed
console.log(string,address,string,string) (contracts/Relayer-flattened.sol#2045-2047) is never used and should be removed
console.log(string,address,string,uint256) (contracts/Relayer-flattened.sol#2041-2043) is never used and should be removed
console.log(string,address,uint256) (contracts/Relayer-flattened.sol#1433-1435) is never used and should be removed
console.log(string,address,uint256,address) (contracts/Relayer-flattened.sol#2037-2039) is never used and should be removed
console.log(string,address,uint256,bool) (contracts/Relayer-flattened.sol#2033-2035) is never used and should be removed
console.log(string,address,uint256,string) (contracts/Relayer-flattened.sol#2029-2031) is never used and should be removed
console.log(string,address,uint256,uint256) (contracts/Relayer-flattened.sol#2025-2027) is never used and should be removed
console.log(string,bool) (contracts/Relayer-flattened.sol#1281-1283) is never used and should be removed
console.log(string,bool,address) (contracts/Relayer-flattened.sol#1429-1431) is never used and should be removed
console.log(string,bool,address,address) (contracts/Relayer-flattened.sol#2021-2023) is never used and should be removed
console.log(string,bool,address,bool) (contracts/Relayer-flattened.sol#2017-2019) is never used and should be removed
console.log(string,bool,address,string) (contracts/Relayer-flattened.sol#2013-2015) is never used and should be removed
console.log(string,bool,address,uint256) (contracts/Relayer-flattened.sol#2009-2011) is never used and should be removed
console.log(string,bool,bool) (contracts/Relayer-flattened.sol#1425-1427) is never used and should be removed
console.log(string,bool,bool,address) (contracts/Relayer-flattened.sol#2005-2007) is never used and should be removed
console.log(string,bool,bool,bool) (contracts/Relayer-flattened.sol#2001-2003) is never used and should be removed
console.log(string,bool,bool,string) (contracts/Relayer-flattened.sol#1997-1999) is never used and should be removed
console.log(string,bool,bool,uint256) (contracts/Relayer-flattened.sol#1993-1995) is never used and should be removed
console.log(string,bool,string) (contracts/Relayer-flattened.sol#1421-1423) is never used and should be removed
console.log(string,bool,string,address) (contracts/Relayer-flattened.sol#1989-1991) is never used and should be removed

```



`console.log(string,bool,string,bool)` (contracts/Relayer-flattened.sol#1985-1987) is never used and should be removed

`console.log(string,bool,string,string)` (contracts/Relayer-flattened.sol#1981-1983) is never used and should be removed

`console.log(string,bool,string,uint256)` (contracts/Relayer-flattened.sol#1977-1979) is never used and should be removed

`console.log(string,bool,uint256)` (contracts/Relayer-flattened.sol#1417-1419) is never used and should be removed

`console.log(string,bool,uint256,address)` (contracts/Relayer-flattened.sol#1973-1975) is never used and should be removed

`console.log(string,bool,uint256,bool)` (contracts/Relayer-flattened.sol#1969-1971) is never used and should be removed

`console.log(string,bool,uint256,string)` (contracts/Relayer-flattened.sol#1965-1967) is never used and should be removed

`console.log(string,bool,uint256,uint256)` (contracts/Relayer-flattened.sol#1961-1963) is never used and should be removed

`console.log(string,string)` (contracts/Relayer-flattened.sol#1277-1279) is never used and should be removed

`console.log(string,string,address)` (contracts/Relayer-flattened.sol#1413-1415) is never used and should be removed

`console.log(string,string,address,address)` (contracts/Relayer-flattened.sol#1957-1959) is never used and should be removed

`console.log(string,string,address,bool)` (contracts/Relayer-flattened.sol#1953-1955) is never used and should be removed

`console.log(string,string,address,string)` (contracts/Relayer-flattened.sol#1949-1951) is never used and should be removed

`console.log(string,string,address,uint256)` (contracts/Relayer-flattened.sol#1945-1947) is never used and should be removed

`console.log(string,string,bool)` (contracts/Relayer-flattened.sol#1409-1411) is never used and should be removed

`console.log(string,string,bool,address)` (contracts/Relayer-flattened.sol#1941-1943) is never used and should be removed

`console.log(string,string,bool,bool)` (contracts/Relayer-flattened.sol#1937-1939) is never used and should be removed

`console.log(string,string,bool,string)` (contracts/Relayer-flattened.sol#1933-1935) is never used and should be removed

`console.log(string,string,bool,uint256)` (contracts/Relayer-flattened.sol#1929-1931) is never used and should be removed

`console.log(string,string,string)` (contracts/Relayer-flattened.sol#1405-1407) is never used and should be removed

`console.log(string,string,string,address)` (contracts/Relayer-flattened.sol#1925-1927) is never used and should be removed

`console.log(string,string,string,bool)` (contracts/Relayer-flattened.sol#1921-1923) is never used and should be removed

`console.log(string,string,string,string)` (contracts/Relayer-flattened.sol#1917-1919) is never used and should be removed

`console.log(string,string,string,uint256)` (contracts/Relayer-flattened.sol#1913-1915) is never used and should be removed

`console.log(string,string,uint256)` (contracts/Relayer-flattened.sol#1401-1403) is never used and should be removed

`console.log(string,string,uint256,address)` (contracts/Relayer-flattened.sol#1909-1911) is never used and should be removed

`console.log(string,string,uint256,bool)` (contracts/Relayer-flattened.sol#1905-1907) is never used and should be removed

`console.log(string,string,uint256,string)` (contracts/Relayer-flattened.sol#1901-1903) is never used and should be removed

`console.log(string,string,uint256,uint256)` (contracts/Relayer-flattened.sol#1897-1899) is never used and should be removed



```

console.log(string,uint256) (contracts/Relayer-flattened.sol#1273-1275) is never
used and should be removed
console.log(string,uint256,address) (contracts/Relayer-flattened.sol#1397-1399) is
never used and should be removed
console.log(string,uint256,address,address) (contracts/Relayer-flattened.sol#1893-
1895) is never used and should be removed
console.log(string,uint256,address,bool) (contracts/Relayer-flattened.sol#1889-
1891) is never used and should be removed
console.log(string,uint256,address,string) (contracts/Relayer-flattened.sol#1885-
1887) is never used and should be removed
console.log(string,uint256,address,uint256) (contracts/Relayer-flattened.sol#1881-
1883) is never used and should be removed
console.log(string,uint256,bool) (contracts/Relayer-flattened.sol#1393-1395) is
never used and should be removed
console.log(string,uint256,bool,address) (contracts/Relayer-flattened.sol#1877-
1879) is never used and should be removed
console.log(string,uint256,bool,bool) (contracts/Relayer-flattened.sol#1873-1875)
is never used and should be removed
console.log(string,uint256,bool,string) (contracts/Relayer-flattened.sol#1869-
1871) is never used and should be removed
console.log(string,uint256,bool,uint256) (contracts/Relayer-flattened.sol#1865-
1867) is never used and should be removed
console.log(string,uint256,string) (contracts/Relayer-flattened.sol#1389-1391) is
never used and should be removed
console.log(string,uint256,string,address) (contracts/Relayer-flattened.sol#1861-
1863) is never used and should be removed
console.log(string,uint256,string,bool) (contracts/Relayer-flattened.sol#1857-
1859) is never used and should be removed
console.log(string,uint256,string,string) (contracts/Relayer-flattened.sol#1853-
1855) is never used and should be removed
console.log(string,uint256,string,uint256) (contracts/Relayer-flattened.sol#1849-
1851) is never used and should be removed
console.log(string,uint256,uint256) (contracts/Relayer-flattened.sol#1385-1387) is
never used and should be removed
console.log(string,uint256,uint256,address) (contracts/Relayer-flattened.sol#1845-
1847) is never used and should be removed
console.log(string,uint256,uint256,bool) (contracts/Relayer-flattened.sol#1841-
1843) is never used and should be removed
console.log(string,uint256,uint256,string) (contracts/Relayer-flattened.sol#1837-
1839) is never used and should be removed
console.log(string,uint256,uint256,uint256) (contracts/Relayer-flattened.sol#1833-
1835) is never used and should be removed
console.log(uint256) (contracts/Relayer-flattened.sol#1241-1243) is never used and
should be removed
console.log(uint256,address) (contracts/Relayer-flattened.sol#1269-1271) is never
used and should be removed
console.log(uint256,address,address) (contracts/Relayer-flattened.sol#1381-1383)
is never used and should be removed
console.log(uint256,address,address,address) (contracts/Relayer-
flattened.sol#1829-1831) is never used and should be removed
console.log(uint256,address,address,bool) (contracts/Relayer-flattened.sol#1825-
1827) is never used and should be removed
console.log(uint256,address,address,string) (contracts/Relayer-flattened.sol#1821-
1823) is never used and should be removed
console.log(uint256,address,address,uint256) (contracts/Relayer-
flattened.sol#1817-1819) is never used and should be removed
console.log(uint256,address,bool) (contracts/Relayer-flattened.sol#1377-1379) is
never used and should be removed

```



console.log(uint256,address,bool,address) (contracts/Relayer-flattened.sol#1813-1815) is never used and should be removed
console.log(uint256,address,bool,bool) (contracts/Relayer-flattened.sol#1809-1811) is never used and should be removed
console.log(uint256,address,bool,string) (contracts/Relayer-flattened.sol#1805-1807) is never used and should be removed
console.log(uint256,address,bool,uint256) (contracts/Relayer-flattened.sol#1801-1803) is never used and should be removed
console.log(uint256,address,string) (contracts/Relayer-flattened.sol#1373-1375) is never used and should be removed
console.log(uint256,address,string,address) (contracts/Relayer-flattened.sol#1797-1799) is never used and should be removed
console.log(uint256,address,string,bool) (contracts/Relayer-flattened.sol#1793-1795) is never used and should be removed
console.log(uint256,address,string,string) (contracts/Relayer-flattened.sol#1789-1791) is never used and should be removed
console.log(uint256,address,string,uint256) (contracts/Relayer-flattened.sol#1785-1787) is never used and should be removed
console.log(uint256,address,uint256) (contracts/Relayer-flattened.sol#1369-1371) is never used and should be removed
console.log(uint256,address,uint256,address) (contracts/Relayer-flattened.sol#1781-1783) is never used and should be removed
console.log(uint256,address,uint256,bool) (contracts/Relayer-flattened.sol#1777-1779) is never used and should be removed
console.log(uint256,address,uint256,string) (contracts/Relayer-flattened.sol#1773-1775) is never used and should be removed
console.log(uint256,address,uint256,uint256) (contracts/Relayer-flattened.sol#1769-1771) is never used and should be removed
console.log(uint256,bool) (contracts/Relayer-flattened.sol#1265-1267) is never used and should be removed
console.log(uint256,bool,address) (contracts/Relayer-flattened.sol#1365-1367) is never used and should be removed
console.log(uint256,bool,address,address) (contracts/Relayer-flattened.sol#1765-1767) is never used and should be removed
console.log(uint256,bool,address,bool) (contracts/Relayer-flattened.sol#1761-1763) is never used and should be removed
console.log(uint256,bool,address,string) (contracts/Relayer-flattened.sol#1757-1759) is never used and should be removed
console.log(uint256,bool,address,uint256) (contracts/Relayer-flattened.sol#1753-1755) is never used and should be removed
console.log(uint256,bool,bool) (contracts/Relayer-flattened.sol#1361-1363) is never used and should be removed
console.log(uint256,bool,bool,address) (contracts/Relayer-flattened.sol#1749-1751) is never used and should be removed
console.log(uint256,bool,bool,bool) (contracts/Relayer-flattened.sol#1745-1747) is never used and should be removed
console.log(uint256,bool,bool,string) (contracts/Relayer-flattened.sol#1741-1743) is never used and should be removed
console.log(uint256,bool,bool,uint256) (contracts/Relayer-flattened.sol#1737-1739) is never used and should be removed
console.log(uint256,bool,string) (contracts/Relayer-flattened.sol#1357-1359) is never used and should be removed
console.log(uint256,bool,string,address) (contracts/Relayer-flattened.sol#1733-1735) is never used and should be removed
console.log(uint256,bool,string,bool) (contracts/Relayer-flattened.sol#1729-1731) is never used and should be removed
console.log(uint256,bool,string,string) (contracts/Relayer-flattened.sol#1725-1727) is never used and should be removed




```

console.log(uint256,bool,string,uint256) (contracts/Relayer-flattened.sol#1721-1723) is never used and should be removed
console.log(uint256,bool,uint256) (contracts/Relayer-flattened.sol#1353-1355) is never used and should be removed
console.log(uint256,bool,uint256,address) (contracts/Relayer-flattened.sol#1717-1719) is never used and should be removed
console.log(uint256,bool,uint256,bool) (contracts/Relayer-flattened.sol#1713-1715) is never used and should be removed
console.log(uint256,bool,uint256,string) (contracts/Relayer-flattened.sol#1709-1711) is never used and should be removed
console.log(uint256,bool,uint256,uint256) (contracts/Relayer-flattened.sol#1705-1707) is never used and should be removed
console.log(uint256,string) (contracts/Relayer-flattened.sol#1261-1263) is never used and should be removed
console.log(uint256,string,address) (contracts/Relayer-flattened.sol#1349-1351) is never used and should be removed
console.log(uint256,string,address,address) (contracts/Relayer-flattened.sol#1701-1703) is never used and should be removed
console.log(uint256,string,address,bool) (contracts/Relayer-flattened.sol#1697-1699) is never used and should be removed
console.log(uint256,string,address,string) (contracts/Relayer-flattened.sol#1693-1695) is never used and should be removed
console.log(uint256,string,address,uint256) (contracts/Relayer-flattened.sol#1689-1691) is never used and should be removed
console.log(uint256,string,bool) (contracts/Relayer-flattened.sol#1345-1347) is never used and should be removed
console.log(uint256,string,bool,address) (contracts/Relayer-flattened.sol#1685-1687) is never used and should be removed
console.log(uint256,string,bool,bool) (contracts/Relayer-flattened.sol#1681-1683) is never used and should be removed
console.log(uint256,string,bool,string) (contracts/Relayer-flattened.sol#1677-1679) is never used and should be removed
console.log(uint256,string,bool,uint256) (contracts/Relayer-flattened.sol#1673-1675) is never used and should be removed
console.log(uint256,string,string) (contracts/Relayer-flattened.sol#1341-1343) is never used and should be removed
console.log(uint256,string,string,address) (contracts/Relayer-flattened.sol#1669-1671) is never used and should be removed
console.log(uint256,string,string,bool) (contracts/Relayer-flattened.sol#1665-1667) is never used and should be removed
console.log(uint256,string,string,string) (contracts/Relayer-flattened.sol#1661-1663) is never used and should be removed
console.log(uint256,string,string,uint256) (contracts/Relayer-flattened.sol#1657-1659) is never used and should be removed
console.log(uint256,string,uint256) (contracts/Relayer-flattened.sol#1337-1339) is never used and should be removed
console.log(uint256,string,uint256,address) (contracts/Relayer-flattened.sol#1653-1655) is never used and should be removed
console.log(uint256,string,uint256,bool) (contracts/Relayer-flattened.sol#1649-1651) is never used and should be removed
console.log(uint256,string,uint256,string) (contracts/Relayer-flattened.sol#1645-1647) is never used and should be removed
console.log(uint256,string,uint256,uint256) (contracts/Relayer-flattened.sol#1641-1643) is never used and should be removed
console.log(uint256,uint256) (contracts/Relayer-flattened.sol#1257-1259) is never used and should be removed
console.log(uint256,uint256,address) (contracts/Relayer-flattened.sol#1333-1335) is never used and should be removed

```



```

console.log(uint256,uint256,address,address) (contracts/Relayer-
flattened.sol#1637-1639) is never used and should be removed
console.log(uint256,uint256,address,bool) (contracts/Relayer-flattened.sol#1633-
1635) is never used and should be removed
console.log(uint256,uint256,address,string) (contracts/Relayer-flattened.sol#1629-
1631) is never used and should be removed
console.log(uint256,uint256,address,uint256) (contracts/Relayer-
flattened.sol#1625-1627) is never used and should be removed
console.log(uint256,uint256,bool) (contracts/Relayer-flattened.sol#1329-1331) is
never used and should be removed
console.log(uint256,uint256,bool,address) (contracts/Relayer-flattened.sol#1621-
1623) is never used and should be removed
console.log(uint256,uint256,bool,bool) (contracts/Relayer-flattened.sol#1617-1619)
is never used and should be removed
console.log(uint256,uint256,bool,string) (contracts/Relayer-flattened.sol#1613-
1615) is never used and should be removed
console.log(uint256,uint256,bool,uint256) (contracts/Relayer-flattened.sol#1609-
1611) is never used and should be removed
console.log(uint256,uint256,string) (contracts/Relayer-flattened.sol#1325-1327) is
never used and should be removed
console.log(uint256,uint256,string,address) (contracts/Relayer-flattened.sol#1605-
1607) is never used and should be removed
console.log(uint256,uint256,string,bool) (contracts/Relayer-flattened.sol#1601-
1603) is never used and should be removed
console.log(uint256,uint256,string,string) (contracts/Relayer-flattened.sol#1597-
1599) is never used and should be removed
console.log(uint256,uint256,string,uint256) (contracts/Relayer-flattened.sol#1593-
1595) is never used and should be removed
console.log(uint256,uint256,uint256) (contracts/Relayer-flattened.sol#1321-1323)
is never used and should be removed
console.log(uint256,uint256,uint256,address) (contracts/Relayer-
flattened.sol#1589-1591) is never used and should be removed
console.log(uint256,uint256,uint256,bool) (contracts/Relayer-flattened.sol#1585-
1587) is never used and should be removed
console.log(uint256,uint256,uint256,string) (contracts/Relayer-flattened.sol#1581-
1583) is never used and should be removed
console.log(uint256,uint256,uint256,uint256) (contracts/Relayer-
flattened.sol#1577-1579) is never used and should be removed
console.logAddress(address) (contracts/Relayer-flattened.sol#1105-1107) is never
used and should be removed
console.logBool(bool) (contracts/Relayer-flattened.sol#1101-1103) is never used
and should be removed
console.logBytes(bytes) (contracts/Relayer-flattened.sol#1109-1111) is never used
and should be removed
console.logBytes1(bytes1) (contracts/Relayer-flattened.sol#1113-1115) is never
used and should be removed
console.logBytes10(bytes10) (contracts/Relayer-flattened.sol#1149-1151) is never
used and should be removed
console.logBytes11(bytes11) (contracts/Relayer-flattened.sol#1153-1155) is never
used and should be removed
console.logBytes12(bytes12) (contracts/Relayer-flattened.sol#1157-1159) is never
used and should be removed
console.logBytes13(bytes13) (contracts/Relayer-flattened.sol#1161-1163) is never
used and should be removed
console.logBytes14(bytes14) (contracts/Relayer-flattened.sol#1165-1167) is never
used and should be removed
console.logBytes15(bytes15) (contracts/Relayer-flattened.sol#1169-1171) is never
used and should be removed

```

```

console.logBytes16(bytes16) (contracts/Relayer-flattened.sol#1173-1175) is never
used and should be removed
console.logBytes17(bytes17) (contracts/Relayer-flattened.sol#1177-1179) is never
used and should be removed
console.logBytes18(bytes18) (contracts/Relayer-flattened.sol#1181-1183) is never
used and should be removed
console.logBytes19(bytes19) (contracts/Relayer-flattened.sol#1185-1187) is never
used and should be removed
console.logBytes2(bytes2) (contracts/Relayer-flattened.sol#1117-1119) is never
used and should be removed
console.logBytes20(bytes20) (contracts/Relayer-flattened.sol#1189-1191) is never
used and should be removed
console.logBytes21(bytes21) (contracts/Relayer-flattened.sol#1193-1195) is never
used and should be removed
console.logBytes22(bytes22) (contracts/Relayer-flattened.sol#1197-1199) is never
used and should be removed
console.logBytes23(bytes23) (contracts/Relayer-flattened.sol#1201-1203) is never
used and should be removed
console.logBytes24(bytes24) (contracts/Relayer-flattened.sol#1205-1207) is never
used and should be removed
console.logBytes25(bytes25) (contracts/Relayer-flattened.sol#1209-1211) is never
used and should be removed
console.logBytes26(bytes26) (contracts/Relayer-flattened.sol#1213-1215) is never
used and should be removed
console.logBytes27(bytes27) (contracts/Relayer-flattened.sol#1217-1219) is never
used and should be removed
console.logBytes28(bytes28) (contracts/Relayer-flattened.sol#1221-1223) is never
used and should be removed
console.logBytes29(bytes29) (contracts/Relayer-flattened.sol#1225-1227) is never
used and should be removed
console.logBytes3(bytes3) (contracts/Relayer-flattened.sol#1121-1123) is never
used and should be removed
console.logBytes30(bytes30) (contracts/Relayer-flattened.sol#1229-1231) is never
used and should be removed
console.logBytes31(bytes31) (contracts/Relayer-flattened.sol#1233-1235) is never
used and should be removed
console.logBytes32(bytes32) (contracts/Relayer-flattened.sol#1237-1239) is never
used and should be removed
console.logBytes4(bytes4) (contracts/Relayer-flattened.sol#1125-1127) is never
used and should be removed
console.logBytes5(bytes5) (contracts/Relayer-flattened.sol#1129-1131) is never
used and should be removed
console.logBytes6(bytes6) (contracts/Relayer-flattened.sol#1133-1135) is never
used and should be removed
console.logBytes7(bytes7) (contracts/Relayer-flattened.sol#1137-1139) is never
used and should be removed
console.logBytes8(bytes8) (contracts/Relayer-flattened.sol#1141-1143) is never
used and should be removed
console.logBytes9(bytes9) (contracts/Relayer-flattened.sol#1145-1147) is never
used and should be removed
console.logInt(int256) (contracts/Relayer-flattened.sol#1089-1091) is never used
and should be removed
console.logString(string) (contracts/Relayer-flattened.sol#1097-1099) is never
used and should be removed
console.logUint(uint256) (contracts/Relayer-flattened.sol#1093-1095) is never used
and should be removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:

```



Pragma version^0.8.0 (contracts/Relayer-flattened.sol#3) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/Relayer-flattened.sol#68) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/Relayer-flattened.sol#154) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/Relayer-flattened.sol#206) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/Relayer-flattened.sol#236) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/Relayer-flattened.sol#278) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/Relayer-flattened.sol#350) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/Relayer-flattened.sol#387) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/Relayer-flattened.sol#687) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version0.8.4 (contracts/Relayer-flattened.sol#952) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/Relayer-flattened.sol#968) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version>=0.4.22<0.9.0 (contracts/Relayer-flattened.sol#1071) is too complex
 Pragma version0.8.4 (contracts/Relayer-flattened.sol#2607) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 solc-0.8.4 is not recommended for deployment
 Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity>
 INFO:Detectors:
 Function IERC20Permit.DOMAIN_SEPARATOR() (contracts/Relayer-flattened.sol#58) is not in mixedCase
 Function ERC165Upgradeable.__ERC165_init() (contracts/Relayer-flattened.sol#256-258) is not in mixedCase
 Function ERC165Upgradeable.__ERC165_init_unchained() (contracts/Relayer-flattened.sol#260-261) is not in mixedCase
 Variable ERC165Upgradeable.__gap (contracts/Relayer-flattened.sol#268) is not in mixedCase
 Function ContextUpgradeable.__Context_init() (contracts/Relayer-flattened.sol#364-366) is not in mixedCase
 Function ContextUpgradeable.__Context_init_unchained() (contracts/Relayer-flattened.sol#368-369) is not in mixedCase
 Variable ContextUpgradeable.__gap (contracts/Relayer-flattened.sol#377) is not in mixedCase
 Function AccessControlUpgradeable.__AccessControl_init() (contracts/Relayer-flattened.sol#748-752) is not in mixedCase
 Function AccessControlUpgradeable.__AccessControl_init_unchained() (contracts/Relayer-flattened.sol#754-755) is not in mixedCase
 Variable AccessControlUpgradeable.__gap (contracts/Relayer-flattened.sol#943) is not in mixedCase
 Function AccessControlEnumerableUpgradeable.__AccessControlEnumerable_init() (contracts/Relayer-flattened.sol#987-992) is not in mixedCase
 Function AccessControlEnumerableUpgradeable.__AccessControlEnumerable_init_unchained() (contracts/Relayer-flattened.sol#994-995) is not in mixedCase
 Variable AccessControlEnumerableUpgradeable.__gap (contracts/Relayer-flattened.sol#1062) is not in mixedCase
 Contract console (contracts/Relayer-flattened.sol#1073-2601) is not in CapWords
 Parameter Relayer.initialize(address,address,IDailyToken)._defaultAdmin (contracts/Relayer-flattened.sol#2619) is not in mixedCase



```
> slither contracts/DailyCopTokenRoot-flattened.sol
```

```
Compilation warnings/errors on contracts/DailyCopTokenRoot-flattened.sol:
Warning: SPDX license identifier not provided in source file. Before publishing,
consider adding a comment containing "SPDX-License-Identifier: <SPDX-License>" to
each source file. Use "SPDX-License-Identifier: UNLICENSED" for non-open-source
code. Please see https://spdx.org for more information.
--> contracts/DailyCopTokenRoot-flattened.sol
```

INFO:Detectors:

```
AccessControlEnumerable.grantRole(bytes32,address) (contracts/DailyCopTokenRoot-  
flattened.sol#1730-1733) ignores return value by _roleMembers[role].add(account)  
(contracts/DailyCopTokenRoot-flattened.sol#1732)  
AccessControlEnumerable.revokeRole(bytes32,address) (contracts/DailyCopTokenRoot-  
flattened.sol#1738-1741) ignores return value by  
_roleMembers[role].remove(account) (contracts/DailyCopTokenRoot-  
flattened.sol#1740)
```



AccessControlEnumerable.renounceRole(bytes32,address)
 (contracts/DailyCopTokenRoot-flattened.sol#1746-1749) ignores return value by
 _roleMembers[role].remove(account) (contracts/DailyCopTokenRoot-
 flattened.sol#1748)
 AccessControlEnumerable._setupRole(bytes32,address) (contracts/DailyCopTokenRoot-
 flattened.sol#1754-1757) ignores return value by _roleMembers[role].add(account)
 (contracts/DailyCopTokenRoot-flattened.sol#1756)
 Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#unused-return>
 INFO:Detectors:
 ERC20Permit.constructor(string).name (contracts/DailyCopTokenRoot-
 flattened.sol#1614) shadows:
 - ERC20.name() (contracts/DailyCopTokenRoot-flattened.sol#515-517)
 (function)
 - IERC20Metadata.name() (contracts/DailyCopTokenRoot-flattened.sol#280)
 (function)
 Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#local-variable-shadowing>
 INFO:Detectors:
 Variable 'ECDSA.recover(bytes32,bytes).r (contracts/DailyCopTokenRoot-
 flattened.sol#35)' in ECDSA.recover(bytes32,bytes) (contracts/DailyCopTokenRoot-
 flattened.sol#30-59) potentially used before declaration: r =
 mload(uint256)(signature + 0x20) (contracts/DailyCopTokenRoot-flattened.sol#52)
 Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#pre-declaration-usage-of-local-variables>
 INFO:Detectors:
 ERC20Permit.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)
 (contracts/DailyCopTokenRoot-flattened.sol#1619-1638) uses timestamp for
 comparisons
 Dangerous comparisons:
 - require(bool,string)(block.timestamp <= deadline,ERC20Permit: expired
 deadline) (contracts/DailyCopTokenRoot-flattened.sol#1628)
 Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#block-timestamp>
 INFO:Detectors:
 ECDSA.recover(bytes32,bytes) (contracts/DailyCopTokenRoot-flattened.sol#30-59)
 uses assembly
 - INLINE ASM (contracts/DailyCopTokenRoot-flattened.sol#40-44)
 - INLINE ASM (contracts/DailyCopTokenRoot-flattened.sol#51-54)
 ECDSA.recover(bytes32,bytes32,bytes32) (contracts/DailyCopTokenRoot-
 flattened.sol#68-80) uses assembly
 - INLINE ASM (contracts/DailyCopTokenRoot-flattened.sol#75-78)
 Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-usage>
 INFO:Detectors:
 Different versions of Solidity is used:
 - Version used: ['0.8.4', '^0.8.0']
 - ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#3)
 - ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#149)
 - ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#237)
 - ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#267)
 - ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#301)
 - ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#350)
 - ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#458)
 - ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#819)
 - ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#885)
 - ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#916)
 - ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#951)
 - ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#1024)



- ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#1325)
- ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#1582)
- ^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#1674)
- 0.8.4 (contracts/DailyCopTokenRoot-flattened.sol#1766)
- 0.8.4 (contracts/DailyCopTokenRoot-flattened.sol#1785)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#different-pragma-directives-are-used>

INFO:Detectors:

AccessControl._setRoleAdmin(bytes32,bytes32) (contracts/DailyCopTokenRoot-flattened.sol#1554-1557) is never used and should be removed

Context._msgData() (contracts/DailyCopTokenRoot-flattened.sol#254-256) is never used and should be removed

Counters.decrement(Counters.Counter) (contracts/DailyCopTokenRoot-flattened.sol#329-335) is never used and should be removed

Counters.reset(Counters.Counter) (contracts/DailyCopTokenRoot-flattened.sol#337-339) is never used and should be removed

ECDSA.recover(bytes32,bytes) (contracts/DailyCopTokenRoot-flattened.sol#30-59) is never used and should be removed

ECDSA.recover(bytes32,bytes32,bytes32) (contracts/DailyCopTokenRoot-flattened.sol#68-80) is never used and should be removed

ECDSA.toEthSignedMessageHash(bytes32) (contracts/DailyCopTokenRoot-flattened.sol#121-125) is never used and should be removed

ERC20._burn(address,uint256) (contracts/DailyCopTokenRoot-flattened.sol#728-743) is never used and should be removed

EnumerableSet.add(EnumerableSet.Bytes32Set,bytes32) (contracts/DailyCopTokenRoot-flattened.sol#1166-1168) is never used and should be removed

EnumerableSet.add(EnumerableSet.UintSet,uint256) (contracts/DailyCopTokenRoot-flattened.sol#1274-1276) is never used and should be removed

EnumerableSet.at(EnumerableSet.Bytes32Set,uint256) (contracts/DailyCopTokenRoot-flattened.sol#1204-1206) is never used and should be removed

EnumerableSet.at(EnumerableSet.UintSet,uint256) (contracts/DailyCopTokenRoot-flattened.sol#1312-1314) is never used and should be removed

EnumerableSet.contains(EnumerableSet.AddressSet,address)

(contracts/DailyCopTokenRoot-flattened.sol#1237-1239) is never used and should be removed

EnumerableSet.contains(EnumerableSet.Bytes32Set,bytes32)

(contracts/DailyCopTokenRoot-flattened.sol#1183-1185) is never used and should be removed

EnumerableSet.contains(EnumerableSet.UintSet,uint256)

(contracts/DailyCopTokenRoot-flattened.sol#1291-1293) is never used and should be removed

EnumerableSet.length(EnumerableSet.Bytes32Set) (contracts/DailyCopTokenRoot-flattened.sol#1190-1192) is never used and should be removed

EnumerableSet.length(EnumerableSet.UintSet) (contracts/DailyCopTokenRoot-flattened.sol#1298-1300) is never used and should be removed

EnumerableSet.remove(EnumerableSet.Bytes32Set,bytes32)

(contracts/DailyCopTokenRoot-flattened.sol#1176-1178) is never used and should be removed

EnumerableSet.remove(EnumerableSet.UintSet,uint256) (contracts/DailyCopTokenRoot-flattened.sol#1284-1286) is never used and should be removed

Strings.toHexString(uint256) (contracts/DailyCopTokenRoot-flattened.sol#987-998) is never used and should be removed

Strings.toString(uint256) (contracts/DailyCopTokenRoot-flattened.sol#962-982) is never used and should be removed

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code>

INFO:Detectors:
Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#3) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6



Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#149) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#237) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#267) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#301) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#350) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#458) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#819) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#885) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#916) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#951) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#1024) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#1325) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#1582) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenRoot-flattened.sol#1674) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version0.8.4 (contracts/DailyCopTokenRoot-flattened.sol#1766) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version0.8.4 (contracts/DailyCopTokenRoot-flattened.sol#1785) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 solc-0.8.4 is not recommended for deployment
 Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity>
 INFO:Detectors:
 Variable EIP712._CACHED_DOMAIN_SEPARATOR (contracts/DailyCopTokenRoot-flattened.sol#377) is not in mixedCase
 Variable EIP712._CACHED_CHAIN_ID (contracts/DailyCopTokenRoot-flattened.sol#378) is not in mixedCase
 Variable EIP712._HASHED_NAME (contracts/DailyCopTokenRoot-flattened.sol#380) is not in mixedCase
 Variable EIP712._HASHED_VERSION (contracts/DailyCopTokenRoot-flattened.sol#381) is not in mixedCase
 Variable EIP712._TYPE_HASH (contracts/DailyCopTokenRoot-flattened.sol#382) is not in mixedCase
 Function IERC20Permit.DOMAIN_SEPARATOR() (contracts/DailyCopTokenRoot-flattened.sol#874) is not in mixedCase
 Function ERC20Permit.DOMAIN_SEPARATOR() (contracts/DailyCopTokenRoot-flattened.sol#1651-1653) is not in mixedCase
 Variable ERC20Permit._PERMIT_TYPEHASH (contracts/DailyCopTokenRoot-flattened.sol#1606-1607) is not in mixedCase
 Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions>
 INFO:Detectors:
 ERC20Permit._PERMIT_TYPEHASH (contracts/DailyCopTokenRoot-flattened.sol#1606-1607) should be constant
 Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant>




```

INFO:Detectors:
name() should be declared external:
  - ERC20.name() (contracts/DailyCopTokenRoot-flattened.sol#515-517)
symbol() should be declared external:
  - ERC20.symbol() (contracts/DailyCopTokenRoot-flattened.sol#523-525)
decimals() should be declared external:
  - ERC20.decimals() (contracts/DailyCopTokenRoot-flattened.sol#540-542)
totalSupply() should be declared external:
  - ERC20.totalSupply() (contracts/DailyCopTokenRoot-flattened.sol#547-549)
balanceOf(address) should be declared external:
  - ERC20.balanceOf(address) (contracts/DailyCopTokenRoot-
flattened.sol#554-556)
transfer(address,uint256) should be declared external:
  - ERC20.transfer(address,uint256) (contracts/DailyCopTokenRoot-
flattened.sol#566-569)
allowance(address,address) should be declared external:
  - ERC20.allowance(address,address) (contracts/DailyCopTokenRoot-
flattened.sol#574-576)
approve(address,uint256) should be declared external:
  - ERC20.approve(address,uint256) (contracts/DailyCopTokenRoot-
flattened.sol#585-588)
transferFrom(address,address,uint256) should be declared external:
  - ERC20.transferFrom(address,address,uint256)
(contracts/DailyCopTokenRoot-flattened.sol#603-617)
increaseAllowance(address,uint256) should be declared external:
  - ERC20.increaseAllowance(address,uint256) (contracts/DailyCopTokenRoot-
flattened.sol#631-634)
decreaseAllowance(address,uint256) should be declared external:
  - ERC20.decreaseAllowance(address,uint256) (contracts/DailyCopTokenRoot-
flattened.sol#650-658)
permit(address,address,uint256,uint256,uint8,bytes32,bytes32) should be declared
external:
  -
ERC20Permit.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)
(contracts/DailyCopTokenRoot-flattened.sol#1619-1638)
nonces(address) should be declared external:
  - ERC20Permit.nonces(address) (contracts/DailyCopTokenRoot-
flattened.sol#1643-1645)
getRoleMember(bytes32,uint256) should be declared external:
  - AccessControlEnumerable.getRoleMember(bytes32,uint256)
(contracts/DailyCopTokenRoot-flattened.sol#1715-1717)
getRoleMemberCount(bytes32) should be declared external:
  - AccessControlEnumerable.getRoleMemberCount(bytes32)
(contracts/DailyCopTokenRoot-flattened.sol#1723-1725)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-
function-that-could-be-declared-external
INFO:Slither:contracts/DailyCopTokenRoot-flattened.sol analyzed (19 contracts with
75 detectors), 73 result(s) found
INFO:Slither:Use https://crytic.io/ to get access to additional detectors and
Github integration

```

DailyCopTokenChild

```
> slither contracts/DailyCopTokenChild-flattened.sol
```

```
Compilation warnings/errors on contracts/DailyCopTokenChild-flattened.sol:
```



Warning: SPDX license identifier not provided in source file. Before publishing, consider adding a comment containing "SPDX-License-Identifier: <SPDX-License>" to each source file. Use "SPDX-License-Identifier: UNLICENSED" for non-open-source code. Please see <https://spdx.org> for more information.

--> contracts/DailyCopTokenChild-flattened.sol

INFO:Detectors:

AccessControlEnumerable.grantRole(bytes32,address) (contracts/DailyCopTokenChild-flattened.sol#1951-1954) ignores return value by _roleMembers[role].add(account) (contracts/DailyCopTokenChild-flattened.sol#1953)

AccessControlEnumerable.revokeRole(bytes32,address) (contracts/DailyCopTokenChild-flattened.sol#1959-1962) ignores return value by _roleMembers[role].remove(account) (contracts/DailyCopTokenChild-flattened.sol#1961)

AccessControlEnumerable.renounceRole(bytes32,address) (contracts/DailyCopTokenChild-flattened.sol#1967-1970) ignores return value by _roleMembers[role].remove(account) (contracts/DailyCopTokenChild-flattened.sol#1969)

AccessControlEnumerable._setupRole(bytes32,address) (contracts/DailyCopTokenChild-flattened.sol#1975-1978) ignores return value by _roleMembers[role].add(account) (contracts/DailyCopTokenChild-flattened.sol#1977)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#unused-return>

INFO:Detectors:

ERC20Permit.constructor(string).name (contracts/DailyCopTokenChild-flattened.sol#2069) shadows:

- ERC20.name() (contracts/DailyCopTokenChild-flattened.sol#357-359)

(function)

- IERC20Metadata.name() (contracts/DailyCopTokenChild-flattened.sol#179)

(function)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#local-variable-shadowing>

INFO:Detectors:

Variable 'ECDSA.recover(bytes32,bytes).r (contracts/DailyCopTokenChild-flattened.sol#1383)' in ECDSA.recover(bytes32,bytes)

(contracts/DailyCopTokenChild-flattened.sol#1378-1407) potentially used before declaration: r = mload(uint256)(signature + 0x20) (contracts/DailyCopTokenChild-flattened.sol#1400)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#pre-declaration-usage-of-local-variables>

INFO:Detectors:

ERC20Permit.permit(address,address,uint256,uint256,uint8,bytes32,bytes32) (contracts/DailyCopTokenChild-flattened.sol#2074-2093) uses timestamp for comparisons

Dangerous comparisons:

- require(bool,string)(block.timestamp <= deadline,ERC20Permit: expired deadline) (contracts/DailyCopTokenChild-flattened.sol#2083)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#block-timestamp>

INFO:Detectors:

ECDSA.recover(bytes32,bytes) (contracts/DailyCopTokenChild-flattened.sol#1378-1407) uses assembly

- INLINE ASM (contracts/DailyCopTokenChild-flattened.sol#1388-1392)

- INLINE ASM (contracts/DailyCopTokenChild-flattened.sol#1399-1402)

ECDSA.recover(bytes32,bytes32,bytes32) (contracts/DailyCopTokenChild-flattened.sol#1416-1428) uses assembly

- INLINE ASM (contracts/DailyCopTokenChild-flattened.sol#1423-1426)



console._sendLogPayload(bytes) (contracts/DailyCopTokenChild-flattened.sol#2134-2141) uses assembly

- INLINE ASM (contracts/DailyCopTokenChild-flattened.sol#2137-2140)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-usage>

INFO:Detectors:

Different versions of Solidity is used:

- Version used: ['0.8.4', '>=0.4.22<0.9.0', '^0.8.0']
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#3)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#90)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#138)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#166)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#199)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#247)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#300)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#660)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#690)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#723)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#795)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1095)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1351)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1497)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1605)
- 0.8.4 (contracts/DailyCopTokenChild-flattened.sol#1670)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1694)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1895)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1989)
- ^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#2037)
- >=0.4.22<0.9.0 (contracts/DailyCopTokenChild-flattened.sol#2129)
- 0.8.4 (contracts/DailyCopTokenChild-flattened.sol#3666)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#different-pragma-directives-are-used>

INFO:Detectors:

AccessControl._setRoleAdmin(bytes32,bytes32) (contracts/DailyCopTokenChild-flattened.sol#1324-1327) is never used and should be removed

Context._msgData() (contracts/DailyCopTokenChild-flattened.sol#155-157) is never used and should be removed

Counters.decrement(Counters.Counter) (contracts/DailyCopTokenChild-flattened.sol#227-233) is never used and should be removed

Counters.reset(Counters.Counter) (contracts/DailyCopTokenChild-flattened.sol#235-237) is never used and should be removed

DailyCopTokenChild._beforeTokenTransfer(address,address,uint256)

(contracts/DailyCopTokenChild-flattened.sol#3754-3760) is never used and should be removed

ECDSA.recover(bytes32,bytes) (contracts/DailyCopTokenChild-flattened.sol#1378-1407) is never used and should be removed

ECDSA.recover(bytes32,bytes32,bytes32) (contracts/DailyCopTokenChild-flattened.sol#1416-1428) is never used and should be removed

ECDSA.toEthSignedMessageHash(bytes32) (contracts/DailyCopTokenChild-flattened.sol#1469-1473) is never used and should be removed

ERC20Snapshot._beforeTokenTransfer(address,address,uint256)

(contracts/DailyCopTokenChild-flattened.sol#1813-1833) is never used and should be removed

ERC20Snapshot._lastSnapshotId(uint256[]) (contracts/DailyCopTokenChild-flattened.sol#1878-1884) is never used and should be removed

ERC20Snapshot._updateAccountSnapshot(address) (contracts/DailyCopTokenChild-flattened.sol#1862-1864) is never used and should be removed



ERC20Snapshot._updateSnapshot(ERC20Snapshot.Snapshots,uint256)
 (contracts/DailyCopTokenChild-flattened.sol#1870-1876) is never used and should be removed
 ERC20Snapshot._updateTotalSupplySnapshot() (contracts/DailyCopTokenChild-flattened.sol#1866-1868) is never used and should be removed
 EnumerableSet.add(EnumerableSet.Bytes32Set,bytes32) (contracts/DailyCopTokenChild-flattened.sol#937-939) is never used and should be removed
 EnumerableSet.add(EnumerableSet.UintSet,uint256) (contracts/DailyCopTokenChild-flattened.sol#1045-1047) is never used and should be removed
 EnumerableSet.at(EnumerableSet.Bytes32Set,uint256) (contracts/DailyCopTokenChild-flattened.sol#975-977) is never used and should be removed
 EnumerableSet.at(EnumerableSet.UintSet,uint256) (contracts/DailyCopTokenChild-flattened.sol#1083-1085) is never used and should be removed
 EnumerableSet.contains(EnumerableSet.AddressSet,address)
 (contracts/DailyCopTokenChild-flattened.sol#1008-1010) is never used and should be removed
 EnumerableSet.contains(EnumerableSet.Bytes32Set,bytes32)
 (contracts/DailyCopTokenChild-flattened.sol#954-956) is never used and should be removed
 EnumerableSet.contains(EnumerableSet.UintSet,uint256)
 (contracts/DailyCopTokenChild-flattened.sol#1062-1064) is never used and should be removed
 EnumerableSet.length(EnumerableSet.Bytes32Set) (contracts/DailyCopTokenChild-flattened.sol#961-963) is never used and should be removed
 EnumerableSet.length(EnumerableSet.UintSet) (contracts/DailyCopTokenChild-flattened.sol#1069-1071) is never used and should be removed
 EnumerableSet.remove(EnumerableSet.Bytes32Set,bytes32)
 (contracts/DailyCopTokenChild-flattened.sol#947-949) is never used and should be removed
 EnumerableSet.remove(EnumerableSet.UintSet,uint256) (contracts/DailyCopTokenChild-flattened.sol#1055-1057) is never used and should be removed
 Math.ceilDiv(uint256,uint256) (contracts/DailyCopTokenChild-flattened.sol#125-128) is never used and should be removed
 Math.max(uint256,uint256) (contracts/DailyCopTokenChild-flattened.sol#99-101) is never used and should be removed
 Math.min(uint256,uint256) (contracts/DailyCopTokenChild-flattened.sol#106-108) is never used and should be removed
 Strings.toHexString(uint256) (contracts/DailyCopTokenChild-flattened.sol#759-770) is never used and should be removed
 Strings.toString(uint256) (contracts/DailyCopTokenChild-flattened.sol#734-754) is never used and should be removed
 console._sendLogPayload(bytes) (contracts/DailyCopTokenChild-flattened.sol#2134-2141) is never used and should be removed
 console.log() (contracts/DailyCopTokenChild-flattened.sol#2143-2145) is never used and should be removed
 console.log(address) (contracts/DailyCopTokenChild-flattened.sol#2311-2313) is never used and should be removed
 console.log(address,address) (contracts/DailyCopTokenChild-flattened.sol#2375-2377) is never used and should be removed
 console.log(address,address,address) (contracts/DailyCopTokenChild-flattened.sol#2631-2633) is never used and should be removed
 console.log(address,address,address,address) (contracts/DailyCopTokenChild-flattened.sol#3655-3657) is never used and should be removed
 console.log(address,address,address,bool) (contracts/DailyCopTokenChild-flattened.sol#3651-3653) is never used and should be removed
 console.log(address,address,address,string) (contracts/DailyCopTokenChild-flattened.sol#3647-3649) is never used and should be removed
 console.log(address,address,address,uint256) (contracts/DailyCopTokenChild-flattened.sol#3643-3645) is never used and should be removed



```

console.log(address,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#2627-2629) is never used and should be removed
console.log(address,address,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#3639-3641) is never used and should be removed
console.log(address,address,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#3635-3637) is never used and should be removed
console.log(address,address,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#3631-3633) is never used and should be removed
console.log(address,address,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3627-3629) is never used and should be removed
console.log(address,address,string) (contracts/DailyCopTokenChild-
flattened.sol#2623-2625) is never used and should be removed
console.log(address,address,string,address) (contracts/DailyCopTokenChild-
flattened.sol#3623-3625) is never used and should be removed
console.log(address,address,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#3619-3621) is never used and should be removed
console.log(address,address,string,string) (contracts/DailyCopTokenChild-
flattened.sol#3615-3617) is never used and should be removed
console.log(address,address,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3611-3613) is never used and should be removed
console.log(address,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2619-2621) is never used and should be removed
console.log(address,address,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#3607-3609) is never used and should be removed
console.log(address,address,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#3603-3605) is never used and should be removed
console.log(address,address,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#3599-3601) is never used and should be removed
console.log(address,address,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3595-3597) is never used and should be removed
console.log(address,bool) (contracts/DailyCopTokenChild-flattened.sol#2371-2373)
is never used and should be removed
console.log(address,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#2615-2617) is never used and should be removed
console.log(address,bool,address,address) (contracts/DailyCopTokenChild-
flattened.sol#3591-3593) is never used and should be removed
console.log(address,bool,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#3587-3589) is never used and should be removed
console.log(address,bool,address,string) (contracts/DailyCopTokenChild-
flattened.sol#3583-3585) is never used and should be removed
console.log(address,bool,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3579-3581) is never used and should be removed
console.log(address,bool,bool) (contracts/DailyCopTokenChild-flattened.sol#2611-
2613) is never used and should be removed
console.log(address,bool,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#3575-3577) is never used and should be removed
console.log(address,bool,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#3571-3573) is never used and should be removed
console.log(address,bool,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#3567-3569) is never used and should be removed
console.log(address,bool,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3563-3565) is never used and should be removed
console.log(address,bool,string) (contracts/DailyCopTokenChild-flattened.sol#2607-
2609) is never used and should be removed
console.log(address,bool,string,address) (contracts/DailyCopTokenChild-
flattened.sol#3559-3561) is never used and should be removed
console.log(address,bool,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#3555-3557) is never used and should be removed

```



```

console.log(address,bool,string,string) (contracts/DailyCopTokenChild-
flattened.sol#3551-3553) is never used and should be removed
console.log(address,bool,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3547-3549) is never used and should be removed
console.log(address,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2603-2605) is never used and should be removed
console.log(address,bool,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#3543-3545) is never used and should be removed
console.log(address,bool,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#3539-3541) is never used and should be removed
console.log(address,bool,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#3535-3537) is never used and should be removed
console.log(address,bool,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3531-3533) is never used and should be removed
console.log(address,string) (contracts/DailyCopTokenChild-flattened.sol#2367-2369)
is never used and should be removed
console.log(address,string,address) (contracts/DailyCopTokenChild-
flattened.sol#2599-2601) is never used and should be removed
console.log(address,string,address,address) (contracts/DailyCopTokenChild-
flattened.sol#3527-3529) is never used and should be removed
console.log(address,string,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#3523-3525) is never used and should be removed
console.log(address,string,address,string) (contracts/DailyCopTokenChild-
flattened.sol#3519-3521) is never used and should be removed
console.log(address,string,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3515-3517) is never used and should be removed
console.log(address,string,bool) (contracts/DailyCopTokenChild-flattened.sol#2595-
2597) is never used and should be removed
console.log(address,string,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#3511-3513) is never used and should be removed
console.log(address,string,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#3507-3509) is never used and should be removed
console.log(address,string,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#3503-3505) is never used and should be removed
console.log(address,string,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3499-3501) is never used and should be removed
console.log(address,string,string) (contracts/DailyCopTokenChild-
flattened.sol#2591-2593) is never used and should be removed
console.log(address,string,string,address) (contracts/DailyCopTokenChild-
flattened.sol#3495-3497) is never used and should be removed
console.log(address,string,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#3491-3493) is never used and should be removed
console.log(address,string,string,string) (contracts/DailyCopTokenChild-
flattened.sol#3487-3489) is never used and should be removed
console.log(address,string,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3483-3485) is never used and should be removed
console.log(address,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2587-2589) is never used and should be removed
console.log(address,string,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#3479-3481) is never used and should be removed
console.log(address,string,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#3475-3477) is never used and should be removed
console.log(address,string,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#3471-3473) is never used and should be removed
console.log(address,string,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3467-3469) is never used and should be removed
console.log(address,uint256) (contracts/DailyCopTokenChild-flattened.sol#2363-
2365) is never used and should be removed

```



```

console.log(address,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#2583-2585) is never used and should be removed
console.log(address,uint256,address,address) (contracts/DailyCopTokenChild-
flattened.sol#3463-3465) is never used and should be removed
console.log(address,uint256,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#3459-3461) is never used and should be removed
console.log(address,uint256,address,string) (contracts/DailyCopTokenChild-
flattened.sol#3455-3457) is never used and should be removed
console.log(address,uint256,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3451-3453) is never used and should be removed
console.log(address,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#2579-2581) is never used and should be removed
console.log(address,uint256,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#3447-3449) is never used and should be removed
console.log(address,uint256,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#3443-3445) is never used and should be removed
console.log(address,uint256,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#3439-3441) is never used and should be removed
console.log(address,uint256,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3435-3437) is never used and should be removed
console.log(address,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#2575-2577) is never used and should be removed
console.log(address,uint256,string,address) (contracts/DailyCopTokenChild-
flattened.sol#3431-3433) is never used and should be removed
console.log(address,uint256,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#3427-3429) is never used and should be removed
console.log(address,uint256,string,string) (contracts/DailyCopTokenChild-
flattened.sol#3423-3425) is never used and should be removed
console.log(address,uint256,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3419-3421) is never used and should be removed
console.log(address,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2571-2573) is never used and should be removed
console.log(address,uint256,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#3415-3417) is never used and should be removed
console.log(address,uint256,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#3411-3413) is never used and should be removed
console.log(address,uint256,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#3407-3409) is never used and should be removed
console.log(address,uint256,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3403-3405) is never used and should be removed
console.log(bool) (contracts/DailyCopTokenChild-flattened.sol#2307-2309) is never
used and should be removed
console.log(bool,address) (contracts/DailyCopTokenChild-flattened.sol#2359-2361)
is never used and should be removed
console.log(bool,address,address) (contracts/DailyCopTokenChild-
flattened.sol#2567-2569) is never used and should be removed
console.log(bool,address,address,address) (contracts/DailyCopTokenChild-
flattened.sol#3399-3401) is never used and should be removed
console.log(bool,address,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#3395-3397) is never used and should be removed
console.log(bool,address,address,string) (contracts/DailyCopTokenChild-
flattened.sol#3391-3393) is never used and should be removed
console.log(bool,address,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3387-3389) is never used and should be removed
console.log(bool,address,bool) (contracts/DailyCopTokenChild-flattened.sol#2563-
2565) is never used and should be removed
console.log(bool,address,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#3383-3385) is never used and should be removed

```




```

console.log(bool,address,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#3379-3381) is never used and should be removed
console.log(bool,address,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#3375-3377) is never used and should be removed
console.log(bool,address,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3371-3373) is never used and should be removed
console.log(bool,address,string) (contracts/DailyCopTokenChild-flattened.sol#2559-
2561) is never used and should be removed
console.log(bool,address,string,address) (contracts/DailyCopTokenChild-
flattened.sol#3367-3369) is never used and should be removed
console.log(bool,address,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#3363-3365) is never used and should be removed
console.log(bool,address,string,string) (contracts/DailyCopTokenChild-
flattened.sol#3359-3361) is never used and should be removed
console.log(bool,address,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3355-3357) is never used and should be removed
console.log(bool,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2555-2557) is never used and should be removed
console.log(bool,address,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#3351-3353) is never used and should be removed
console.log(bool,address,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#3347-3349) is never used and should be removed
console.log(bool,address,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#3343-3345) is never used and should be removed
console.log(bool,address,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3339-3341) is never used and should be removed
console.log(bool,bool) (contracts/DailyCopTokenChild-flattened.sol#2355-2357) is
never used and should be removed
console.log(bool,bool,address) (contracts/DailyCopTokenChild-flattened.sol#2551-
2553) is never used and should be removed
console.log(bool,bool,address,address) (contracts/DailyCopTokenChild-
flattened.sol#3335-3337) is never used and should be removed
console.log(bool,bool,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#3331-3333) is never used and should be removed
console.log(bool,bool,address,string) (contracts/DailyCopTokenChild-
flattened.sol#3327-3329) is never used and should be removed
console.log(bool,bool,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3323-3325) is never used and should be removed
console.log(bool,bool,bool) (contracts/DailyCopTokenChild-flattened.sol#2547-2549)
is never used and should be removed
console.log(bool,bool,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#3319-3321) is never used and should be removed
console.log(bool,bool,bool,bool) (contracts/DailyCopTokenChild-flattened.sol#3315-
3317) is never used and should be removed
console.log(bool,bool,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#3311-3313) is never used and should be removed
console.log(bool,bool,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3307-3309) is never used and should be removed
console.log(bool,bool,string) (contracts/DailyCopTokenChild-flattened.sol#2543-
2545) is never used and should be removed
console.log(bool,bool,string,address) (contracts/DailyCopTokenChild-
flattened.sol#3303-3305) is never used and should be removed
console.log(bool,bool,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#3299-3301) is never used and should be removed
console.log(bool,bool,string,string) (contracts/DailyCopTokenChild-
flattened.sol#3295-3297) is never used and should be removed
console.log(bool,bool,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3291-3293) is never used and should be removed

```



```

console.log(bool,bool,uint256) (contracts/DailyCopTokenChild-flattened.sol#2539-
2541) is never used and should be removed
console.log(bool,bool,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#3287-3289) is never used and should be removed
console.log(bool,bool,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#3283-3285) is never used and should be removed
console.log(bool,bool,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#3279-3281) is never used and should be removed
console.log(bool,bool,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3275-3277) is never used and should be removed
console.log(bool,string) (contracts/DailyCopTokenChild-flattened.sol#2351-2353) is
never used and should be removed
console.log(bool,string,address) (contracts/DailyCopTokenChild-flattened.sol#2535-
2537) is never used and should be removed
console.log(bool,string,address,address) (contracts/DailyCopTokenChild-
flattened.sol#3271-3273) is never used and should be removed
console.log(bool,string,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#3267-3269) is never used and should be removed
console.log(bool,string,address,string) (contracts/DailyCopTokenChild-
flattened.sol#3263-3265) is never used and should be removed
console.log(bool,string,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3259-3261) is never used and should be removed
console.log(bool,string,bool) (contracts/DailyCopTokenChild-flattened.sol#2531-
2533) is never used and should be removed
console.log(bool,string,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#3255-3257) is never used and should be removed
console.log(bool,string,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#3251-3253) is never used and should be removed
console.log(bool,string,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#3247-3249) is never used and should be removed
console.log(bool,string,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3243-3245) is never used and should be removed
console.log(bool,string,string) (contracts/DailyCopTokenChild-flattened.sol#2527-
2529) is never used and should be removed
console.log(bool,string,string,address) (contracts/DailyCopTokenChild-
flattened.sol#3239-3241) is never used and should be removed
console.log(bool,string,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#3235-3237) is never used and should be removed
console.log(bool,string,string,string) (contracts/DailyCopTokenChild-
flattened.sol#3231-3233) is never used and should be removed
console.log(bool,string,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3227-3229) is never used and should be removed
console.log(bool,string,uint256) (contracts/DailyCopTokenChild-flattened.sol#2523-
2525) is never used and should be removed
console.log(bool,string,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#3223-3225) is never used and should be removed
console.log(bool,string,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#3219-3221) is never used and should be removed
console.log(bool,string,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#3215-3217) is never used and should be removed
console.log(bool,string,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3211-3213) is never used and should be removed
console.log(bool,uint256) (contracts/DailyCopTokenChild-flattened.sol#2347-2349)
is never used and should be removed
console.log(bool,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#2519-2521) is never used and should be removed
console.log(bool,uint256,address,address) (contracts/DailyCopTokenChild-
flattened.sol#3207-3209) is never used and should be removed

```



```

console.log(bool,uint256,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#3203-3205) is never used and should be removed
console.log(bool,uint256,address,string) (contracts/DailyCopTokenChild-
flattened.sol#3199-3201) is never used and should be removed
console.log(bool,uint256,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3195-3197) is never used and should be removed
console.log(bool,uint256,bool) (contracts/DailyCopTokenChild-flattened.sol#2515-
2517) is never used and should be removed
console.log(bool,uint256,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#3191-3193) is never used and should be removed
console.log(bool,uint256,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#3187-3189) is never used and should be removed
console.log(bool,uint256,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#3183-3185) is never used and should be removed
console.log(bool,uint256,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3179-3181) is never used and should be removed
console.log(bool,uint256,string) (contracts/DailyCopTokenChild-flattened.sol#2511-
2513) is never used and should be removed
console.log(bool,uint256,string,address) (contracts/DailyCopTokenChild-
flattened.sol#3175-3177) is never used and should be removed
console.log(bool,uint256,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#3171-3173) is never used and should be removed
console.log(bool,uint256,string,string) (contracts/DailyCopTokenChild-
flattened.sol#3167-3169) is never used and should be removed
console.log(bool,uint256,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3163-3165) is never used and should be removed
console.log(bool,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2507-2509) is never used and should be removed
console.log(bool,uint256,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#3159-3161) is never used and should be removed
console.log(bool,uint256,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#3155-3157) is never used and should be removed
console.log(bool,uint256,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#3151-3153) is never used and should be removed
console.log(bool,uint256,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3147-3149) is never used and should be removed
console.log(string) (contracts/DailyCopTokenChild-flattened.sol#2303-2305) is
never used and should be removed
console.log(string,address) (contracts/DailyCopTokenChild-flattened.sol#2343-2345)
is never used and should be removed
console.log(string,address,address) (contracts/DailyCopTokenChild-
flattened.sol#2503-2505) is never used and should be removed
console.log(string,address,address,address) (contracts/DailyCopTokenChild-
flattened.sol#3143-3145) is never used and should be removed
console.log(string,address,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#3139-3141) is never used and should be removed
console.log(string,address,address,string) (contracts/DailyCopTokenChild-
flattened.sol#3135-3137) is never used and should be removed
console.log(string,address,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3131-3133) is never used and should be removed
console.log(string,address,bool) (contracts/DailyCopTokenChild-flattened.sol#2499-
2501) is never used and should be removed
console.log(string,address,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#3127-3129) is never used and should be removed
console.log(string,address,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#3123-3125) is never used and should be removed
console.log(string,address,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#3119-3121) is never used and should be removed

```




```

console.log(string,address,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3115-3117) is never used and should be removed
console.log(string,address,string) (contracts/DailyCopTokenChild-
flattened.sol#2495-2497) is never used and should be removed
console.log(string,address,string,address) (contracts/DailyCopTokenChild-
flattened.sol#3111-3113) is never used and should be removed
console.log(string,address,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#3107-3109) is never used and should be removed
console.log(string,address,string,string) (contracts/DailyCopTokenChild-
flattened.sol#3103-3105) is never used and should be removed
console.log(string,address,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3099-3101) is never used and should be removed
console.log(string,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2491-2493) is never used and should be removed
console.log(string,address,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#3095-3097) is never used and should be removed
console.log(string,address,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#3091-3093) is never used and should be removed
console.log(string,address,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#3087-3089) is never used and should be removed
console.log(string,address,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3083-3085) is never used and should be removed
console.log(string,bool) (contracts/DailyCopTokenChild-flattened.sol#2339-2341) is
never used and should be removed
console.log(string,bool,address) (contracts/DailyCopTokenChild-flattened.sol#2487-
2489) is never used and should be removed
console.log(string,bool,address,address) (contracts/DailyCopTokenChild-
flattened.sol#3079-3081) is never used and should be removed
console.log(string,bool,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#3075-3077) is never used and should be removed
console.log(string,bool,address,string) (contracts/DailyCopTokenChild-
flattened.sol#3071-3073) is never used and should be removed
console.log(string,bool,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3067-3069) is never used and should be removed
console.log(string,bool,bool) (contracts/DailyCopTokenChild-flattened.sol#2483-
2485) is never used and should be removed
console.log(string,bool,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#3063-3065) is never used and should be removed
console.log(string,bool,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#3059-3061) is never used and should be removed
console.log(string,bool,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#3055-3057) is never used and should be removed
console.log(string,bool,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3051-3053) is never used and should be removed
console.log(string,bool,string) (contracts/DailyCopTokenChild-flattened.sol#2479-
2481) is never used and should be removed
console.log(string,bool,string,address) (contracts/DailyCopTokenChild-
flattened.sol#3047-3049) is never used and should be removed
console.log(string,bool,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#3043-3045) is never used and should be removed
console.log(string,bool,string,string) (contracts/DailyCopTokenChild-
flattened.sol#3039-3041) is never used and should be removed
console.log(string,bool,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3035-3037) is never used and should be removed
console.log(string,bool,uint256) (contracts/DailyCopTokenChild-flattened.sol#2475-
2477) is never used and should be removed
console.log(string,bool,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#3031-3033) is never used and should be removed

```



```

console.log(string,bool,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#3027-3029) is never used and should be removed
console.log(string,bool,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#3023-3025) is never used and should be removed
console.log(string,bool,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3019-3021) is never used and should be removed
console.log(string,string) (contracts/DailyCopTokenChild-flattened.sol#2335-2337)
is never used and should be removed
console.log(string,string,address) (contracts/DailyCopTokenChild-
flattened.sol#2471-2473) is never used and should be removed
console.log(string,string,address,address) (contracts/DailyCopTokenChild-
flattened.sol#3015-3017) is never used and should be removed
console.log(string,string,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#3011-3013) is never used and should be removed
console.log(string,string,address,string) (contracts/DailyCopTokenChild-
flattened.sol#3007-3009) is never used and should be removed
console.log(string,string,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3003-3005) is never used and should be removed
console.log(string,string,bool) (contracts/DailyCopTokenChild-flattened.sol#2467-
2469) is never used and should be removed
console.log(string,string,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#2999-3001) is never used and should be removed
console.log(string,string,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#2995-2997) is never used and should be removed
console.log(string,string,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#2991-2993) is never used and should be removed
console.log(string,string,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2987-2989) is never used and should be removed
console.log(string,string,string) (contracts/DailyCopTokenChild-
flattened.sol#2463-2465) is never used and should be removed
console.log(string,string,string,address) (contracts/DailyCopTokenChild-
flattened.sol#2983-2985) is never used and should be removed
console.log(string,string,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#2979-2981) is never used and should be removed
console.log(string,string,string,string) (contracts/DailyCopTokenChild-
flattened.sol#2975-2977) is never used and should be removed
console.log(string,string,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2971-2973) is never used and should be removed
console.log(string,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2459-2461) is never used and should be removed
console.log(string,string,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#2967-2969) is never used and should be removed
console.log(string,string,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#2963-2965) is never used and should be removed
console.log(string,string,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#2959-2961) is never used and should be removed
console.log(string,string,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2955-2957) is never used and should be removed
console.log(string,uint256) (contracts/DailyCopTokenChild-flattened.sol#2331-2333)
is never used and should be removed
console.log(string,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#2455-2457) is never used and should be removed
console.log(string,uint256,address,address) (contracts/DailyCopTokenChild-
flattened.sol#2951-2953) is never used and should be removed
console.log(string,uint256,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#2947-2949) is never used and should be removed
console.log(string,uint256,address,string) (contracts/DailyCopTokenChild-
flattened.sol#2943-2945) is never used and should be removed

```



```

console.log(string,uint256,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2939-2941) is never used and should be removed
console.log(string,uint256,bool) (contracts/DailyCopTokenChild-flattened.sol#2451-
2453) is never used and should be removed
console.log(string,uint256,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#2935-2937) is never used and should be removed
console.log(string,uint256,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#2931-2933) is never used and should be removed
console.log(string,uint256,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#2927-2929) is never used and should be removed
console.log(string,uint256,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2923-2925) is never used and should be removed
console.log(string,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#2447-2449) is never used and should be removed
console.log(string,uint256,string,address) (contracts/DailyCopTokenChild-
flattened.sol#2919-2921) is never used and should be removed
console.log(string,uint256,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#2915-2917) is never used and should be removed
console.log(string,uint256,string,string) (contracts/DailyCopTokenChild-
flattened.sol#2911-2913) is never used and should be removed
console.log(string,uint256,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2907-2909) is never used and should be removed
console.log(string,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2443-2445) is never used and should be removed
console.log(string,uint256,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#2903-2905) is never used and should be removed
console.log(string,uint256,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#2899-2901) is never used and should be removed
console.log(string,uint256,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#2895-2897) is never used and should be removed
console.log(string,uint256,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2891-2893) is never used and should be removed
console.log(uint256) (contracts/DailyCopTokenChild-flattened.sol#2299-2301) is
never used and should be removed
console.log(uint256,address) (contracts/DailyCopTokenChild-flattened.sol#2327-
2329) is never used and should be removed
console.log(uint256,address,address) (contracts/DailyCopTokenChild-
flattened.sol#2439-2441) is never used and should be removed
console.log(uint256,address,address,address) (contracts/DailyCopTokenChild-
flattened.sol#2887-2889) is never used and should be removed
console.log(uint256,address,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#2883-2885) is never used and should be removed
console.log(uint256,address,address,string) (contracts/DailyCopTokenChild-
flattened.sol#2879-2881) is never used and should be removed
console.log(uint256,address,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2875-2877) is never used and should be removed
console.log(uint256,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#2435-2437) is never used and should be removed
console.log(uint256,address,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#2871-2873) is never used and should be removed
console.log(uint256,address,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#2867-2869) is never used and should be removed
console.log(uint256,address,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#2863-2865) is never used and should be removed
console.log(uint256,address,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2859-2861) is never used and should be removed
console.log(uint256,address,string) (contracts/DailyCopTokenChild-
flattened.sol#2431-2433) is never used and should be removed

```



```

console.log(uint256,address,string,address) (contracts/DailyCopTokenChild-
flattened.sol#2855-2857) is never used and should be removed
console.log(uint256,address,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#2851-2853) is never used and should be removed
console.log(uint256,address,string,string) (contracts/DailyCopTokenChild-
flattened.sol#2847-2849) is never used and should be removed
console.log(uint256,address,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2843-2845) is never used and should be removed
console.log(uint256,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2427-2429) is never used and should be removed
console.log(uint256,address,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#2839-2841) is never used and should be removed
console.log(uint256,address,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#2835-2837) is never used and should be removed
console.log(uint256,address,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#2831-2833) is never used and should be removed
console.log(uint256,address,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2827-2829) is never used and should be removed
console.log(uint256,bool) (contracts/DailyCopTokenChild-flattened.sol#2323-2325)
is never used and should be removed
console.log(uint256,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#2423-2425) is never used and should be removed
console.log(uint256,bool,address,address) (contracts/DailyCopTokenChild-
flattened.sol#2823-2825) is never used and should be removed
console.log(uint256,bool,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#2819-2821) is never used and should be removed
console.log(uint256,bool,address,string) (contracts/DailyCopTokenChild-
flattened.sol#2815-2817) is never used and should be removed
console.log(uint256,bool,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2811-2813) is never used and should be removed
console.log(uint256,bool,bool) (contracts/DailyCopTokenChild-flattened.sol#2419-
2421) is never used and should be removed
console.log(uint256,bool,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#2807-2809) is never used and should be removed
console.log(uint256,bool,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#2803-2805) is never used and should be removed
console.log(uint256,bool,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#2799-2801) is never used and should be removed
console.log(uint256,bool,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2795-2797) is never used and should be removed
console.log(uint256,bool,string) (contracts/DailyCopTokenChild-flattened.sol#2415-
2417) is never used and should be removed
console.log(uint256,bool,string,address) (contracts/DailyCopTokenChild-
flattened.sol#2791-2793) is never used and should be removed
console.log(uint256,bool,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#2787-2789) is never used and should be removed
console.log(uint256,bool,string,string) (contracts/DailyCopTokenChild-
flattened.sol#2783-2785) is never used and should be removed
console.log(uint256,bool,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2779-2781) is never used and should be removed
console.log(uint256,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2411-2413) is never used and should be removed
console.log(uint256,bool,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#2775-2777) is never used and should be removed
console.log(uint256,bool,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#2771-2773) is never used and should be removed
console.log(uint256,bool,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#2767-2769) is never used and should be removed

```




```

console.log(uint256,bool,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2763-2765) is never used and should be removed
console.log(uint256,string) (contracts/DailyCopTokenChild-flattened.sol#2319-2321)
is never used and should be removed
console.log(uint256,string,address) (contracts/DailyCopTokenChild-
flattened.sol#2407-2409) is never used and should be removed
console.log(uint256,string,address,address) (contracts/DailyCopTokenChild-
flattened.sol#2759-2761) is never used and should be removed
console.log(uint256,string,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#2755-2757) is never used and should be removed
console.log(uint256,string,address,string) (contracts/DailyCopTokenChild-
flattened.sol#2751-2753) is never used and should be removed
console.log(uint256,string,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2747-2749) is never used and should be removed
console.log(uint256,string,bool) (contracts/DailyCopTokenChild-flattened.sol#2403-
2405) is never used and should be removed
console.log(uint256,string,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#2743-2745) is never used and should be removed
console.log(uint256,string,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#2739-2741) is never used and should be removed
console.log(uint256,string,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#2735-2737) is never used and should be removed
console.log(uint256,string,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2731-2733) is never used and should be removed
console.log(uint256,string,string) (contracts/DailyCopTokenChild-
flattened.sol#2399-2401) is never used and should be removed
console.log(uint256,string,string,address) (contracts/DailyCopTokenChild-
flattened.sol#2727-2729) is never used and should be removed
console.log(uint256,string,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#2723-2725) is never used and should be removed
console.log(uint256,string,string,string) (contracts/DailyCopTokenChild-
flattened.sol#2719-2721) is never used and should be removed
console.log(uint256,string,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2715-2717) is never used and should be removed
console.log(uint256,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2395-2397) is never used and should be removed
console.log(uint256,string,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#2711-2713) is never used and should be removed
console.log(uint256,string,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#2707-2709) is never used and should be removed
console.log(uint256,string,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#2703-2705) is never used and should be removed
console.log(uint256,string,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2699-2701) is never used and should be removed
console.log(uint256,uint256) (contracts/DailyCopTokenChild-flattened.sol#2315-
2317) is never used and should be removed
console.log(uint256,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#2391-2393) is never used and should be removed
console.log(uint256,uint256,address,address) (contracts/DailyCopTokenChild-
flattened.sol#2695-2697) is never used and should be removed
console.log(uint256,uint256,address,bool) (contracts/DailyCopTokenChild-
flattened.sol#2691-2693) is never used and should be removed
console.log(uint256,uint256,address,string) (contracts/DailyCopTokenChild-
flattened.sol#2687-2689) is never used and should be removed
console.log(uint256,uint256,address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2683-2685) is never used and should be removed
console.log(uint256,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#2387-2389) is never used and should be removed

```



```

console.log(uint256,uint256,bool,address) (contracts/DailyCopTokenChild-
flattened.sol#2679-2681) is never used and should be removed
console.log(uint256,uint256,bool,bool) (contracts/DailyCopTokenChild-
flattened.sol#2675-2677) is never used and should be removed
console.log(uint256,uint256,bool,string) (contracts/DailyCopTokenChild-
flattened.sol#2671-2673) is never used and should be removed
console.log(uint256,uint256,bool,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2667-2669) is never used and should be removed
console.log(uint256,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#2383-2385) is never used and should be removed
console.log(uint256,uint256,string,address) (contracts/DailyCopTokenChild-
flattened.sol#2663-2665) is never used and should be removed
console.log(uint256,uint256,string,bool) (contracts/DailyCopTokenChild-
flattened.sol#2659-2661) is never used and should be removed
console.log(uint256,uint256,string,string) (contracts/DailyCopTokenChild-
flattened.sol#2655-2657) is never used and should be removed
console.log(uint256,uint256,string,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2651-2653) is never used and should be removed
console.log(uint256,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2379-2381) is never used and should be removed
console.log(uint256,uint256,uint256,address) (contracts/DailyCopTokenChild-
flattened.sol#2647-2649) is never used and should be removed
console.log(uint256,uint256,uint256,bool) (contracts/DailyCopTokenChild-
flattened.sol#2643-2645) is never used and should be removed
console.log(uint256,uint256,uint256,string) (contracts/DailyCopTokenChild-
flattened.sol#2639-2641) is never used and should be removed
console.log(uint256,uint256,uint256,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2635-2637) is never used and should be removed
console.logAddress(address) (contracts/DailyCopTokenChild-flattened.sol#2163-2165)
is never used and should be removed
console.logBool(bool) (contracts/DailyCopTokenChild-flattened.sol#2159-2161) is
never used and should be removed
console.logBytes(bytes) (contracts/DailyCopTokenChild-flattened.sol#2167-2169) is
never used and should be removed
console.logBytes1(bytes1) (contracts/DailyCopTokenChild-flattened.sol#2171-2173)
is never used and should be removed
console.logBytes10(bytes10) (contracts/DailyCopTokenChild-flattened.sol#2207-2209)
is never used and should be removed
console.logBytes11(bytes11) (contracts/DailyCopTokenChild-flattened.sol#2211-2213)
is never used and should be removed
console.logBytes12(bytes12) (contracts/DailyCopTokenChild-flattened.sol#2215-2217)
is never used and should be removed
console.logBytes13(bytes13) (contracts/DailyCopTokenChild-flattened.sol#2219-2221)
is never used and should be removed
console.logBytes14(bytes14) (contracts/DailyCopTokenChild-flattened.sol#2223-2225)
is never used and should be removed
console.logBytes15(bytes15) (contracts/DailyCopTokenChild-flattened.sol#2227-2229)
is never used and should be removed
console.logBytes16(bytes16) (contracts/DailyCopTokenChild-flattened.sol#2231-2233)
is never used and should be removed
console.logBytes17(bytes17) (contracts/DailyCopTokenChild-flattened.sol#2235-2237)
is never used and should be removed
console.logBytes18(bytes18) (contracts/DailyCopTokenChild-flattened.sol#2239-2241)
is never used and should be removed
console.logBytes19(bytes19) (contracts/DailyCopTokenChild-flattened.sol#2243-2245)
is never used and should be removed
console.logBytes2(bytes2) (contracts/DailyCopTokenChild-flattened.sol#2175-2177)
is never used and should be removed

```



```

console.logBytes20(bytes20) (contracts/DailyCopTokenChild-flattened.sol#2247-2249)
is never used and should be removed
console.logBytes21(bytes21) (contracts/DailyCopTokenChild-flattened.sol#2251-2253)
is never used and should be removed
console.logBytes22(bytes22) (contracts/DailyCopTokenChild-flattened.sol#2255-2257)
is never used and should be removed
console.logBytes23(bytes23) (contracts/DailyCopTokenChild-flattened.sol#2259-2261)
is never used and should be removed
console.logBytes24(bytes24) (contracts/DailyCopTokenChild-flattened.sol#2263-2265)
is never used and should be removed
console.logBytes25(bytes25) (contracts/DailyCopTokenChild-flattened.sol#2267-2269)
is never used and should be removed
console.logBytes26(bytes26) (contracts/DailyCopTokenChild-flattened.sol#2271-2273)
is never used and should be removed
console.logBytes27(bytes27) (contracts/DailyCopTokenChild-flattened.sol#2275-2277)
is never used and should be removed
console.logBytes28(bytes28) (contracts/DailyCopTokenChild-flattened.sol#2279-2281)
is never used and should be removed
console.logBytes29(bytes29) (contracts/DailyCopTokenChild-flattened.sol#2283-2285)
is never used and should be removed
console.logBytes3(bytes3) (contracts/DailyCopTokenChild-flattened.sol#2179-2181)
is never used and should be removed
console.logBytes30(bytes30) (contracts/DailyCopTokenChild-flattened.sol#2287-2289)
is never used and should be removed
console.logBytes31(bytes31) (contracts/DailyCopTokenChild-flattened.sol#2291-2293)
is never used and should be removed
console.logBytes32(bytes32) (contracts/DailyCopTokenChild-flattened.sol#2295-2297)
is never used and should be removed
console.logBytes4(bytes4) (contracts/DailyCopTokenChild-flattened.sol#2183-2185)
is never used and should be removed
console.logBytes5(bytes5) (contracts/DailyCopTokenChild-flattened.sol#2187-2189)
is never used and should be removed
console.logBytes6(bytes6) (contracts/DailyCopTokenChild-flattened.sol#2191-2193)
is never used and should be removed
console.logBytes7(bytes7) (contracts/DailyCopTokenChild-flattened.sol#2195-2197)
is never used and should be removed
console.logBytes8(bytes8) (contracts/DailyCopTokenChild-flattened.sol#2199-2201)
is never used and should be removed
console.logBytes9(bytes9) (contracts/DailyCopTokenChild-flattened.sol#2203-2205)
is never used and should be removed
console.logInt(int256) (contracts/DailyCopTokenChild-flattened.sol#2147-2149) is
never used and should be removed
console.logString(string) (contracts/DailyCopTokenChild-flattened.sol#2155-2157)
is never used and should be removed
console.logUint(uint256) (contracts/DailyCopTokenChild-flattened.sol#2151-2153) is
never used and should be removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:
Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#3) necessitates a
version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#90) necessitates
a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#138) necessitates
a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#166) necessitates
a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#199) necessitates
a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6

```



Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#247) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#300) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#660) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#690) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#723) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#795) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1095) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1351) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1497) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1605) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version0.8.4 (contracts/DailyCopTokenChild-flattened.sol#1670) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1694) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1895) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#1989) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version^0.8.0 (contracts/DailyCopTokenChild-flattened.sol#2037) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 Pragma version>=0.4.22<0.9.0 (contracts/DailyCopTokenChild-flattened.sol#2129) is too complex
 Pragma version0.8.4 (contracts/DailyCopTokenChild-flattened.sol#3666) necessitates a version too recent to be trusted. Consider deploying with 0.6.12/0.7.6
 solc-0.8.4 is not recommended for deployment
 Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity>
 INFO:Detectors:
 Variable EIP712._CACHED_DOMAIN_SEPARATOR (contracts/DailyCopTokenChild-flattened.sol#1524) is not in mixedCase
 Variable EIP712._CACHED_CHAIN_ID (contracts/DailyCopTokenChild-flattened.sol#1525) is not in mixedCase
 Variable EIP712._HASHED_NAME (contracts/DailyCopTokenChild-flattened.sol#1527) is not in mixedCase
 Variable EIP712._HASHED_VERSION (contracts/DailyCopTokenChild-flattened.sol#1528) is not in mixedCase
 Variable EIP712._TYPE_HASH (contracts/DailyCopTokenChild-flattened.sol#1529) is not in mixedCase
 Function IERC20Permit.DOMAIN_SEPARATOR() (contracts/DailyCopTokenChild-flattened.sol#1660) is not in mixedCase



Function ERC20Permit.DOMAIN_SEPARATOR() (contracts/DailyCopTokenChild-flattened.sol#2106-2108) is not in mixedCase

Variable ERC20Permit._PERMIT_TYPEHASH (contracts/DailyCopTokenChild-flattened.sol#2061-2062) is not in mixedCase

Contract console (contracts/DailyCopTokenChild-flattened.sol#2131-3659) is not in CapWords

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-solidity-naming-conventions>

INFO:Detectors:

console.slitherConstructorConstantVariables() (contracts/DailyCopTokenChild-flattened.sol#2131-3659) uses literals with too many digits:

- `CONSOLE_ADDRESS = address(0x00)` (contracts/DailyCopTokenChild-flattened.sol#2132)

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-digits>

INFO:Detectors:

ERC20Permit._PERMIT_TYPEHASH (contracts/DailyCopTokenChild-flattened.sol#2061-2062) should be constant

Reference: <https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant>

INFO:Detectors:

`name()` should be declared external:

- `ERC20.name()` (contracts/DailyCopTokenChild-flattened.sol#357-359)

`symbol()` should be declared external:

- `ERC20.symbol()` (contracts/DailyCopTokenChild-flattened.sol#365-367)

`decimals()` should be declared external:

- `ERC20.decimals()` (contracts/DailyCopTokenChild-flattened.sol#382-384)

`transfer(address,uint256)` should be declared external:

- `ERC20.transfer(address,uint256)` (contracts/DailyCopTokenChild-flattened.sol#408-411)

`approve(address,uint256)` should be declared external:

- `ERC20.approve(address,uint256)` (contracts/DailyCopTokenChild-flattened.sol#427-430)

`transferFrom(address,address,uint256)` should be declared external:

- `ERC20.transferFrom(address,address,uint256)` (contracts/DailyCopTokenChild-flattened.sol#445-459)

`increaseAllowance(address,uint256)` should be declared external:

- `ERC20.increaseAllowance(address,uint256)` (contracts/DailyCopTokenChild-flattened.sol#473-476)

`decreaseAllowance(address,uint256)` should be declared external:

- `ERC20.decreaseAllowance(address,uint256)` (contracts/DailyCopTokenChild-flattened.sol#492-500)

`balanceOfAt(address,uint256)` should be declared external:

- `ERC20Snapshot.balanceOfAt(address,uint256)` (contracts/DailyCopTokenChild-flattened.sol#1796-1800)

`totalSupplyAt(uint256)` should be declared external:

- `ERC20Snapshot.totalSupplyAt(uint256)` (contracts/DailyCopTokenChild-flattened.sol#1805-1809)

`getRoleMember(bytes32,uint256)` should be declared external:

- `AccessControlEnumerable.getRoleMember(bytes32,uint256)` (contracts/DailyCopTokenChild-flattened.sol#1936-1938)

`getRoleMemberCount(bytes32)` should be declared external:

- `AccessControlEnumerable.getRoleMemberCount(bytes32)` (contracts/DailyCopTokenChild-flattened.sol#1944-1946)

`burn(uint256)` should be declared external:

- `ERC20Burnable.burn(uint256)` (contracts/DailyCopTokenChild-flattened.sol#2005-2007)

`burnFrom(address,uint256)` should be declared external:



```

- ERC20Burnable.burnFrom(address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#2020-2027)
permit(address,address,uint256,uint256,uint8,bytes32,bytes32) should be declared
external:
-
ERC20Permit.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)
(contracts/DailyCopTokenChild-flattened.sol#2074-2093)
nonces(address) should be declared external:
- ERC20Permit.nonces(address) (contracts/DailyCopTokenChild-
flattened.sol#2098-2100)
mint(address,uint256) should be declared external:
- DailyCopTokenChild.mint(address,uint256) (contracts/DailyCopTokenChild-
flattened.sol#3709-3711)
snapshot() should be declared external:
- DailyCopTokenChild.snapshot() (contracts/DailyCopTokenChild-
flattened.sol#3713-3715)
layer1Supply() should be declared external:
- DailyCopTokenChild.layer1Supply() (contracts/DailyCopTokenChild-
flattened.sol#3750-3752)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-
function-that-could-be-declared-external
INFO:Slither:contracts/DailyCopTokenChild-flattened.sol analyzed (24 contracts
with 75 detectors), 473 result(s) found
INFO:Slither:Use https://crytic.io/ to get access to additional detectors and
Github integration

```

