Tokamak Network - TON Staking V2

Security Assessment

2025. 03. 13

Carl Park @4000d

Disclaimer

This report should not be considered to ensure the investment of a particular team or project, or the suitability of a business model.

This report should not be used for decision making to invest or participate in a particular project.

This report identifies vulnerabilities that were overlooked during development and areas requiring additional security measures. However, this does not mean discovering all vulnerabilities and does not guarantee that the source code is completely secure, even if no vulnerabilities are found. This do not provide a complete guarantee of all vulnerabilities and types of attacks that exist even after auditing is complete.

Overview

Github Repository: https://github.com/tokamak-network/ton-staking-v2

Branch	Commit Hash
ton-staking-v2.5	01e198130b178757dd194bd8726a1ab678fca167
fix-l2-seigs-v2	a7760e19801970c63005eecd05ab78c583e15476

Project Summary

Details of the Staking V2.5 upgrade in the Tokamak Network are summarized in Deploy TON staking V2.5 contract + upgrade DAO to support preliminary security council features.

Changes to the TON issuance policy related to this upgrade are summarized in Changes in TON Staking V2.5.

Changes to the DAOCommiteeProxy contract are summarized in What's changing in DAOCommitteeProxy.

Audit Summary

Delivery Date: 2025. 03. 13

Timeline: 2025. 02. 03 – 2025. 03. 13

Vulnerability Summary

Total Findings: 21

Severity	Total	Pending	Declined	Acknowledged	Partially Resolved	Migrated	Resolved
Informational	14	0	0	1	0	0	13
Minor	1	0	0	0	0	0	1
Medium	4	0	0	0	0	0	4
Major	0	0	0	0	0	0	0
Critical	2	0	0	0	0	1	1

Files In Scope

- contracts/proxy/DAOCommitteeProxy2.sol
- contracts/dao/StorageStateCommittee.sol
- contracts/proxy/ProxyStorage2.sol
- contracts/dao/DAOCommittee_V1.sol
- contracts/dao/lib/Agenda.sol
- contracts/dao/DAOCommitteeOwner.sol
- contracts/dao/StorageStateCommitteeV2.sol
- contracts/stake/managers/SeigManagerV1_3.sol
- contracts/proxy/ProxyStorage.sol
- contracts/common/AuthControlSeigManager.sol
- contracts/common/AuthRoleSeigManager.sol
- contracts/stake/managers/SeigManagerStorage.sol
- contracts/stake/managers/SeigManagerV1_1Storage.sol
- contracts/stake/managers/SeigManagerV1_3Storage.sol

- contracts/stake/managers/DepositManagerV1_1.sol
- contracts/common/AccessibleCommon.sol
- contracts/stake/managers/DepositManagerStorage.sol
- contracts/stake/managers/DepositManagerV1_1Storage.sol
- contracts/layer2/L1BridgeRegistryV1_1.sol
- contracts/common/AuthControlL1BridgeRegistry.sol
- contracts/common/AuthRoleL1BridgeRegistry.sol
- contracts/layer2/L1BridgeRegistryStorage.sol
- contracts/layer2/Layer2ManagerV1_1.sol
- contracts/layer2/Layer2ManagerStorage.sol
- contracts/layer2/OperatorManagerV1_1.sol
- contracts/layer2/OperatorManagerStorage.sol
- contracts/dao/CandidateAddOnV1_1.sol
- contracts/dao/CandidateStorage.sol
- contracts/dao/CandidateAddOnStorage.sol

List Of Findings

ID	Severity	Title	Status
TS-1	Critical	Storage layout conflict (DAOCommitteeProxy2)	Resolved
TS-2	Critical	Centralized execution of agenda	Migrated
TS-3	Medium	Duplicate proxy pausing mechanism (DAOCommitteeProxy2)	Resolved
TS-4	Medium	Multiple init possible (CandidateAddOnV1_1)	Resolved
TS-6	Medium	Implementation function conflict (SeigManagerV1_3, SeigManagerV1_2)	Resolved
TS-7	Informational	Multiple SLOAD (getSelectorImplementation2)	Resolved
TS-8	Minor	Cannot disable functions once set (DAOCommitteeProxy2)	Resolved
TS-10	Informational	No event emitted	Resolved
TS-11	Informational	Unnecessary public function	Resolved
TS-12	Informational	Unused internal functions	Resolved
TS-14	Informational	Duplicate code (StorageStateCommittee)	Resolved
TS-15	Informational	Validate user input (CandidateAddOnV1_1.initialize)	Acknowledged
TS-16	Informational	Validate user input (address)	Resolved
TS-17	Informational	Duplicate code (unstakedSeig)	Resolved
TS-18	Informational	Event emitted before state change	Resolved
TS-19	Informational	Duplicate code in modifier (nonRejected)	Resolved
TS-20	Informational	Typo (TOON)	Resolved

TS-21	Informational	Use reason string instead of predefined error (onlySeigniorageCommittee)	Resolved
TS-22	Medium	Infinite length of array in memory	Resolved
TS-23	Informational	Unused variable (_isSenderOperator)	Resolved
TS-24	Informational	Duplicate code (isPauseL2Seigniorage(msg.sender))	Resolved

TS-1 - Storage layout conflict (DAOCommitteeProxy2)

ID	Type	Severity	Location	Status
TS-1	Logical Issue	Critical	contracts/proxy/DAOCommitteeProxy2.sol#L13	Resolved

Description

DAOCommitteeProxy contract(0xDD9f0cCc044B0781289Ee318e5971b0139602C26) deployed on Mainnet uses DAOCommittee_V1 contract (0xdf2ecda32970db7db3428fc12bc1697098418815) as it's implementation contract.

According to the test file(test/layer2/units/6.dao-staking-v2.5.mainnet.test.ts), contracts are upgraded as follows:

- Upgrade existing DAOCommitteeProxy contract's implementation to DAOCommitteeProxy2 contract
- Use 2 new contracts(DAOCommittee_V1, DAOCommitteeOwner) as the implementation contracts of DAOCommitteeProxy2 contract.

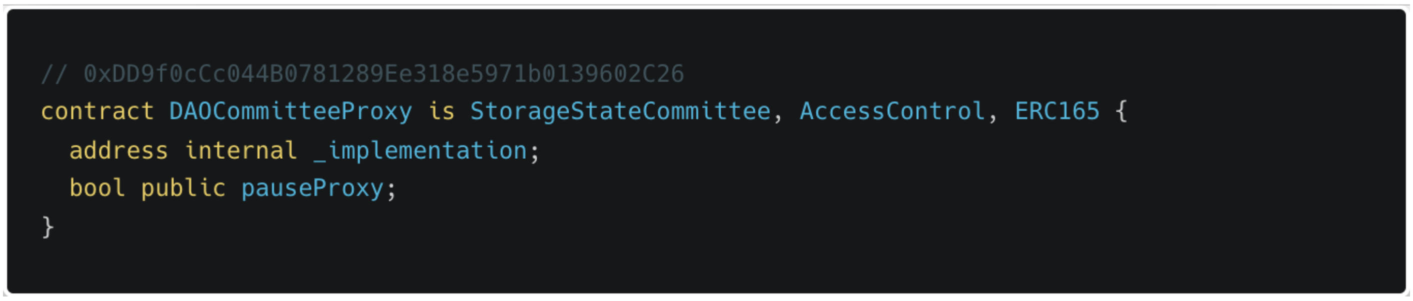
Therefore, the functions executed by the DAOCommitteeProxy contract are executed in the order of DAOCommitteeProxy -> DAOCommiteeProxy2 -> DAOCommitee_V1 or DAOCommiteeOwner.

- DAOCommitteeProxy: Load the implementation contract address from _implementation(Slot#14) state variable.
- DAOCommitteeProxy2: Load the implementation contract address from the proxyImplementation (Slot#14) variable or selectorImplementation (Slot#16) variable, depending on msg.sig..

However, the storage layout of the existing contracts (DAOCommiteeProxy, DAOCommitee_V1) conflicts with and the new contracts (DAOCommiteeProxy2, DAOCommiteeOwner, DAOCommitee_V1) from Slot#14.

1. Existing _implementation, pauseProxy variables conflict with new proxyImplementation.
2. Because the new _implementation uses a different Slot than the existing _implementation, it returns a different value depending on the contract.
3. Because the new pauseProxy uses a different Slot than the existing pauseProxy, it returns a different value depending on the contract.
4. Existing _oldCandidateInfos conflicts with new aliveImplementation.
5. The new _oldCandidateInfos is located in Slot#18 and cannot access existing state variables.
6. The existing wton conflicts with selectorImplementation.
7. Since the new wton is located in slot#19, it returns zero address immediately after the upgrade.

Existing deployed contracts



```
// 0xDD9f0cCc044B0781289Ee318e5971b0139602C26
contract DAOCommitteeProxy is StorageStateCommittee, AccessControl, ERC165 {
  address internal _implementation;
  bool public pauseProxy;
}
```

Slot	Byte Range	Varible Name	Description
			contract StorageStateCommittee storage start
0	0x00 - 0x13	ton	address ton
1	0x00 - 0x13	daoVault	IDAOVault daoVault
2	0x00 - 0x13	agendaManager	IDAOAgendaManager agendaManager
3	0x00 - 0x13	candidateFactory	ICandidateFactory candidateFactory
4	0x00 - 0x13	layer2Registry	ILayer2Registry layer2Registry
5	0x00 - 0x13	seigManager	ISeigManager seigManager

6	0x00 - 0x1F	candidates	length of candidates
7	0x00 - 0x1F	members	length of members
8	0x00 - 0x1F	maxMember	uint256 maxMember
9	0x00 - 0x1F	_candidateInfos	mapping(address => CandidateInfo) _candidateInfos
10	0x00 - 0x1F	quorum	uint256 quorum
11	0x00 - 0x1F	activityRewardPerSecond	uint256 activityRewardPerSecond
			contract StorageStateCommittee storage end
			contract AccessControl storage start
12	0x00 - 0x1F	_roles	mapping (bytes32 => RoleData) _roles
			contract AccessControl storage end
			contract ERC165 storage start
13	0x00 - 0x1F	_supportedInterfaces	mapping(bytes4 => bool) _supportedInterfaces
			contract ERC165 storage end
			contract DAOCommitteeProxy storage start
14	0x00 - 0x13	_implementation	address _implementation
14	0x14 - 0x14	pauseProxy	bool pauseProxy
			contract DAOCommitteeProxy storage end
16	0x00 - 0x13	wton	address wton
			contract StorageStateCommitteeV2 storage end

```
// 0xdf2ecda32970db7db3428fc12bc1697098418815
contract DAOCommittee_V1 is StorageStateCommittee, AccessControl, ERC165A,
StorageStateCommitteeV2 {
  // no additional storage
}
```

Slot	Byte Range	Varible Name	Description
			contract StorageStateCommittee storage start
0	0x00 - 0x13	ton	address ton
1	0x00 - 0x13	daoVault	IDAOVault daoVault
2	0x00 - 0x13	agendaManager	IDAOAgendaManager agendaManager
3	0x00 - 0x13	candidateFactory	ICandidateFactory candidateFactory
4	0x00 - 0x13	layer2Registry	ILayer2Registry layer2Registry
5	0x00 - 0x13	seigManager	ISeigManager seigManager
6	0x00 - 0x1F	candidates	length of candidates
7	0x00 - 0x1F	members	length of members
8	0x00 - 0x1F	maxMember	uint256 maxMember
9	0x00 - 0x1F	_candidateInfos	mapping(address => CandidateInfo) _candidateInfos
10	0x00 - 0x1F	quorum	uint256 quorum
11	0x00 - 0x1F	activityRewardPerSecond	uint256 activityRewardPerSecond
			contract StorageStateCommittee storage end
			contract AccessControl storage start
12	0x00 - 0x1F	_roles	mapping (bytes32 => RoleData) _roles
			contract AccessControl storage end
			contract ERC165 storage start
13	0x00 - 0x1F	_supportedInterfaces	mapping(bytes4 => bool) _supportedInterfaces
			contract ERC165 storage end
			contract StorageStateCommitteeV2 storage start
14	0x00 - 0x13	_implementation	address _implementation
14	0x14 - 0x14	pauseProxy	bool pauseProxy
15	0x00 - 0x1F	_oldCandidateInfos	mapping(address => CandidateInfo2) _oldCandidateInfos
16	0x00 - 0x13	wton	address wton
			contract StorageStateCommitteeV2 storage end

New contracts to deploy

```
contract DAOCommitteeProxy2 is StorageStateCommittee, AccessControl, ERC165A, ProxyStorage2
{
  address internal _implementation;
  bool public pauseProxy;
}
```

Slot	Byte Range	Varible Name	Description
			contract StorageStateCommittee storage start
0	0x00 - 0x13	ton	address ton
1	0x00 - 0x13	daoVault	IDAOVault daoVault
2	0x00 - 0x13	agendaManager	IDAOAgendaManager agendaManager
3	0x00 - 0x13	candidateFactory	ICandidateFactory candidateFactory
4	0x00 - 0x13	layer2Registry	ILayer2Registry layer2Registry
5	0x00 - 0x13	seigManager	ISeigManager seigManager
6	0x00 - 0x1F	candidates	length of candidates
7	0x00 - 0x1F	members	length of members
8	0x00 - 0x1F	maxMember	uint256 maxMember
9	0x00 - 0x1F	_candidateInfos	mapping(address => CandidateInfo) _candidateInfos
10	0x00 - 0x1F	quorum	uint256 quorum
11	0x00 - 0x1F	activityRewardPerSecond	uint256 activityRewardPerSecond
			contract StorageStateCommittee storage end
			contract AccessControl storage start
12	0x00 - 0x1F	_roles	mapping (bytes32 => RoleData) _roles
			contract AccessControl storage end
			contract ERC165 storage start
13	0x00 - 0x1F	_supportedInterfaces	mapping(bytes4 => bool) _supportedInterfaces

			contract ERC165 storage end
			contract ProxyStorage2 storage start
14	0x00 - 0x1F	proxyImplementation	mapping(uint256 => address) proxyImplementation
15	0x00 - 0x1F	aliveImplementation	mapping(address => bool) aliveImplementation
16	0x00 - 0x1F	selectorImplementation	mapping(bytes4 => address) selectorImplementation
			contract ProxyStorage2 storage end
			contract DAOCommitteeProxy2 storage start
17	0x00 - 0x13	_implementation	address _implementation
17	0x14 - 0x14	pauseProxy	bool pauseProxy
			contract DAOCommitteeProxy2 storage end

```
// DAOCommittee_V1 와 DAOCommitteeOwner 는 동일한 Storage layout을 가짐
contract DAOCommittee_V1 is
  StorageStateCommittee,
  AccessControl,
  ERC165A,
  ProxyStorage2,
  StorageStateCommitteeV2
{
  // no additional storage
}
```

Slot	Byte Range	Varible Name	Description
			contract StorageStateCommittee storage start
0	0x00 - 0x13	ton	address ton
1	0x00 - 0x13	daoVault	IDAOVault daoVault
2	0x00 - 0x13	agendaManager	IDAOAgendaManager agendaManager
3	0x00 - 0x13	candidateFactory	ICandidateFactory candidateFactory
4	0x00 - 0x13	layer2Registry	ILayer2Registry layer2Registry
5	0x00 - 0x13	seigManager	ISeigManager seigManager
6	0x00 - 0x1F	candidates	length of candidates

7	0x00 - 0x1F	members	length of members
8	0x00 - 0x1F	maxMember	uint256 maxMember
9	0x00 - 0x1F	_candidateInfos	mapping(address => CandidateInfo) _candidateInfos
10	0x00 - 0x1F	quorum	uint256 quorum
11	0x00 - 0x1F	activityRewardPerSecond	uint256 activityRewardPerSecond
			contract StorageStateCommittee storage end
			contract AccessControl storage start
12	0x00 - 0x1F	_roles	mapping (bytes32 => RoleData) _roles
			contract AccessControl storage end
			contract ERC165 storage start
13	0x00 - 0x1F	_supportedInterfaces	mapping(bytes4 => bool) _supportedInterfaces
			contract ERC165 storage end
			contract ProxyStorage2 storage start
14	0x00 - 0x1F	proxyImplementation	mapping(uint256 => address) proxyImplementation
15	0x00 - 0x1F	aliveImplementation	mapping(address => bool) aliveImplementation
16	0x00 - 0x1F	selectorImplementation	mapping(bytes4 => address) selectorImplementation
			contract ProxyStorage2 storage end
			contract StorageStateCommitteeV2 storage start
17	0x00 - 0x13	_implementation	address _implementation
17	0x14 - 0x14	pauseProxy	bool pauseProxy
18	0x14 - 0x14	_oldCandidateInfos	mapping(address => CandidateInfo2
19	0x14 - 0x14	wton	address wton
20	0x14 - 0x14	layer2Manager	address layer2Manager
21	0x14 - 0x14	candidateAddOnFactory	address candidateAddOnFactory
			contract StorageStateCommitteeV2 storage end

Recommendation

Configure the storage layout of the existing DAOCommittedeeProxy, existing DAOCommittedee_V1, new DAOCommittedee_V1, DAOCommittedeeProxy2, and DAOCommittedeeOwner contracts as follows.

Slot	Byte Range	Varible Name	Description	Contracts
0	0x00 - 0x13	ton	address ton	All
1	0x00 - 0x13	daoVault	IDAOVault daoVault	All
2	0x00 - 0x13	agendaManager	IDAOAgendaManager agendaManager	All
3	0x00 - 0x13	candidateFactory	ICandidateFactory candidateFactory	All
4	0x00 - 0x13	layer2Registry	ILayer2Registry layer2Registry	All
5	0x00 - 0x13	seigManager	ISeigManager seigManager	All
6	0x00 - 0x1F	candidates	length of candidates	All
7	0x00 - 0x1F	members	length of members	All
8	0x00 - 0x1F	maxMember	uint256 maxMember	All
9	0x00 - 0x1F	_candidateInfos	mapping(address => CandidateInfo) _candidateInfos	All
10	0x00 - 0x1F	quorum	uint256 quorum	All
11	0x00 - 0x1F	activityRewardPerSecond	uint256 activityRewardPerSecond	All
12	0x00 - 0x1F	_roles	mapping (bytes32 => RoleData) _roles	All
13	0x00 - 0x1F	_supportedInterfaces	mapping(bytes4 => bool) _supportedInterfaces	All
14	0x00 - 0x13	_implementation	address _implementation	All
14	0x14 - 0x14	pauseProxy	bool pauseProxy	All
15	0x00 - 0x1F	_oldCandidateInfos	mapping(address => CandidateInfo2) _oldCandidateInfos	existing DAOCommittee_V1, new DAOCommittee_V1, DAOCommitteeProxy2, DAOCommitteeOwner
16	0x00 - 0x13	wton	address wton	existing DAOCommittee_V1, new DAOCommittee_V1, DAOCommitteeProxy2, DAOCommitteeOwner

17	0x00 - 0x1F	proxyImplementation	mapping(uint256 => address) proxyImplementation	new DAOCommittee_V1, DAOCommitteeProxy2, DAOCommitteeOwner
18	0x00 - 0x1F	aliveImplementation	mapping(address => bool) aliveImplementation	new DAOCommittee_V1, DAOCommitteeProxy2, DAOCommitteeOwner
19	0x00 - 0x1F	selectorImplementation	mapping(bytes4 => address) selectorImplementation	new DAOCommittee_V1, DAOCommitteeProxy2, DAOCommitteeOwner
20	0x14 - 0x14	layer2Manager	address layer2Manager	DAOCommitteeProxy2, DAOCommitteeOwner
21	0x14 - 0x14	candidateAddOnFactory	address candidateAddOnFactory	DAOCommitteeProxy2, DAOCommitteeOwner

Alleviations

Fixed in the commit: 1462678133ab422114821c0feb194a8eafc5b5b4

TS-2 - Centralized execution of agenda

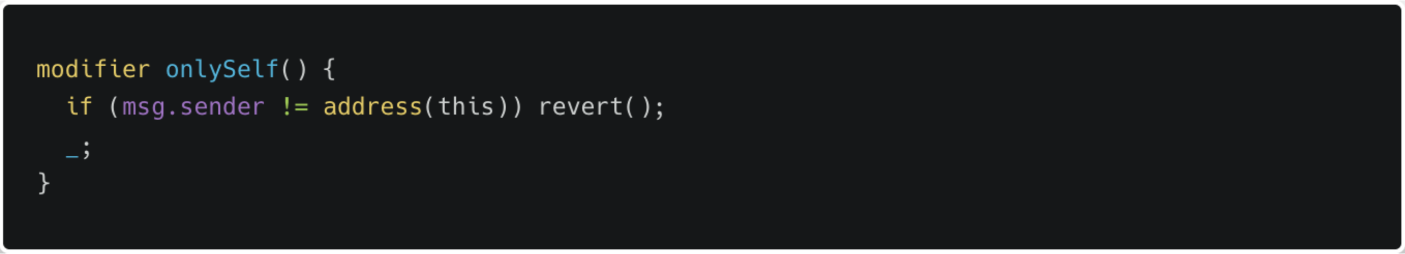
ID	Type	Severity	Location	Status
TS-2	Centralization	Critical	contracts/dao/DAOCommittee_V1.sol#L509-L539	Migrated

Description

The validity of executeAgenda function is determined by voting on an agenda and the function runs arbitrary calls defined for that agenda. However, accounts with DEFAULT_ADMIN_ROLE can run the setAgendaStatus function to change a particular agenda to a valid state regardless of the vote result.
In particular, if the private key of EOA 0xB4983DA083A5118C903910DB4f5a480B1D9f3687 with DEFAULT_ADMIN_ROLE is stolen, there is a possibility that all funds will be stolen.

Recommendation

Remove the setAgendaStatus function so that the agenda can only be run by voting on it. If the team needs an agenda to run the setAgendaStatus function, change the setAgendaStatus function so that it can run only by itself (DAOCommitteeProxy). (e.g., onlySelf)



```
modifier onlySelf() {
  if (msg.sender != address(this)) revert();
  _;
}
```

Alleviations

The team said that they will replace the EOA with the permission with Multisig Wallet contract.

TS-3 - Duplicate proxy pausing mechanism (DAOCommitteeProxy2)

ID	Type	Severity	Location	Status
TS-3	Control Flow	Medium	contracts/proxy/ DAOCommitteeProxy2.sol#L125	Resolved

Description

The 'pauseProxy' state variable is used to distinguish whether the proxy contract's proxy function has been stopped. However, the 'pauseProxy' state variable referred to by the two contracts has different slots inside the existing DAOCommiteeProxy contract (TS-1).

This not only uses SLOAD to increase gas consumption but also harms the readability of contract source codes; it also causes the inefficiency of the contract's management points being added.

Recommendation

1. If the team want to manage the two proxy pause functions separately, DAOCommiteeProxy2 must implement the pause function through separate state variables. Add additional state variables and functions (for example, pauseProxy2, setProxyPause2()).

2. If the team want to use the proxy pause function only in the existing DAOCommiteeProxy contract, delete the proxy pause function in the DAOCommiteeProxy2 contract.

Alleviations

TS-1's fix commit has caused the pauseProxy state variable to use the same slot for both contracts (existing DAOCommiteeProxy and DAOCommiteeProxy2).

Also, the team removed the proxy pause function from the DAOCommiteeProxy2 contract in the commit below.
945643cc8e7666f9357e411115bdb9cd47e15940

TS-4 - Multiple init possible (CandidateAddOnV1_1)

ID	Type	Severity	Location	Status
TS-4	Logical Issue	Medium	contracts/dao/ CandidateAddOnV1_1.sol#L41	Resolved

Description

CandidateAddOnV1_1 contract can have the initialize function called multiple times by the owner. This has the potential to change the major state variables of the contract, such as candidate, isLayer2Candidate, committee, seigManager, memo, ton, wton, to new values.

Recommendation

To prevent this, verify that the value of the state variables are zero value.

Alleviations

Fixed in the commit: b8435ae3bc08bdc9c1b715843e057ba84d25008d

TS-6 - Implementation function conflict (SeigManagerV1_3, SeigManagerV1_2)

ID	Type	Severity	Location	Status
TS-6	Volatile Code	Medium	contracts/stake/managers/SeigManagerV1_2.sol#L429 contracts/stake/managers/SeigManagerV1_2.sol#L510 contracts/stake/managers/SeigManagerV1_2.sol#L564 contracts/stake/managers/SeigManagerV1_2.sol#L568 contracts/stake/managers/SeigManagerV1_2.sol#L606 contracts/stake/managers/SeigManagerV1_2.sol#L614 contracts/stake/managers/SeigManagerV1_2.sol#L628 contracts/stake/managers/SeigManagerV1_3.sol#L224 contracts/stake/managers/SeigManagerV1_3.sol#L347 contracts/stake/managers/SeigManagerV1_3.sol#L380 contracts/stake/managers/SeigManagerV1_3.sol#L384 contracts/stake/managers/SeigManagerV1_3.sol#L388 contracts/stake/managers/SeigManagerV1_3.sol#L397 contracts/stake/managers/SeigManagerV1_3.sol#L697	Resolved

Description

SeigManagerV1_3 contract is one of the implementation contracts of SeigManagerProxy contract and is used with SeigManagerV1_2 contract. Since the SeigManagerProxy contract uses the SeigManagerV1_3 contract as the logic contract for some functions only, it does not need to be implemented by other logic contracts.

In particular, implementing the same function differently not only increases gas consumption during contract deployment, but also reduces code readability and makes it difficult to find actual implementation contracts, making it difficult to manage proxy contracts.

Recommendation

Duplicate external functions should be implemented in only one logic contract. If the team cannot delete a function due to the interface, the team can specify in the contract source code that the function does not run the same way as reverted ("implemented in SeigManagerV1_3").

Alleviations

Fixed in the commit: e8f414ab4406e3bb232460b6db0e73294fd23ae1

TS-7 - Multiple SLOAD (getSelectorImplementation2)

ID	Type	Severity	Location	Status
TS-7	Gas Optimization	Informational	contracts/proxy/DAOCommitteeProxy2.sol#L96-L99	Resolved

Description

The contract runs selectorImplementation[_selector] many times to read the same state variable every time it checks each conditional statement. In the worst case scenario, it uses three SLOADs.

Recommendation

Using selectorImplementation[_selector] in a local variable allows memory to be used to reduce gas consumption.

Alleviations

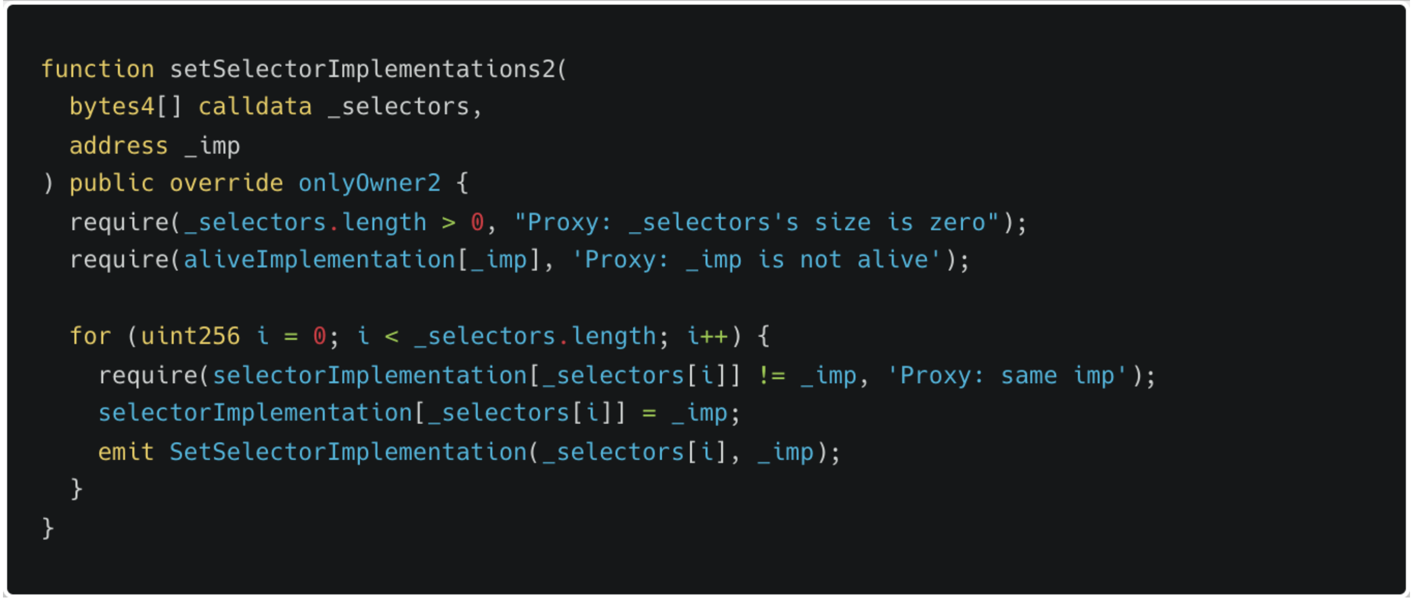
Fixed in the commit: 6600a0849106a1d141fb6da207ba2a296fa0bbba

TS-8 - Cannot disable functions once set (DAOCommitteeProxy2)

ID	Type	Severity	Location	Status
TS-8	Coding Style	Minor	contracts/proxy/ DAOCommitteeProxy2.sol#L64	Resolved

Description

The setSelectorImplementations2 function sets the implementation contract for the functions corresponding to the _selectors to _imp.



```
function setSelectorImplementations2(
  bytes4[] calldata _selectors,
  address _imp
) public override onlyOwner2 {
  require(_selectors.length > 0, "Proxy: _selectors's size is zero");
  require(aliveImplementation[_imp], 'Proxy: _imp is not alive');

  for (uint256 i = 0; i < _selectors.length; i++) {
    require(selectorImplementation[_selectors[i]] != _imp, 'Proxy: same imp');
    selectorImplementation[_selectors[i]] = _imp;
    emit SetSelectorImplementation(_selectors[i], _imp);
  }
}
```

The team may need to deactivate the registered functions during the contract upgrade process. To remove the registered function again, following transactions must be executed.

1. setAliveImplementation2(0x00, true)
2. setSelectorImplementations2(SELECTORS_TO_REMOVE, 0x00)

This method requires additional gas consumption, as it requires additional unnecessary functions to be executed, and registering non-contract addresses with aliveImplementation can also make the code less readable because it is out of context for the corresponding state variables.

Recommendation

If it is intended to remove registered functions using a method such as setAliveImplementation2 (0x00, true), add the relevant information to the comment of the setSelectorImplementations2 function and the aliveImplementation variable.

Otherwise, add an `unset` version of the setSelectorImplementations2 function for flexible contract maintenance.

Alleviations

Fixed in the commit: 061f2a2dbae3bf377446965873ff38d14b44a95d

TS-10 - No event emitted

ID	Type	Severity	Location	Status
TS-10	Volatile Code	Informational	contracts/dao/Candidate.sol#L81 contracts/dao/Candidate.sol#L88 contracts/dao/Candidate.sol#L95 contracts/dao/DAOCommitteeOwner.sol#L85 contracts/dao/DAOCommitteeOwner.sol#L93 contracts/dao/DAOCommitteeOwner.sol#L167 contracts/dao/DAOCommitteeOwner.sol#L173 contracts/dao/DAOCommitteeOwner.sol#L177 contracts/dao/DAOCommitteeOwner.sol#L181 contracts/stake/managers/SeigManagerV1_3.sol#L159 contracts/stake/managers/SeigManagerV1_3.sol#L167 contracts/stake/managers/SeigManagerV1_3.sol#L175 contracts/stake/managers/SeigManagerV1_3.sol#L179	Resolved

Description

The mentioned functions change the important state variable of the contract, but do not emit an event.

Recommendation

Define the appropriate events and emit them.

Alleviations

Fixed in below commits:

- 58b898bc66ad59f3a9990f168ca23885c63bf1c1
- 31568419aeef34777c0289473174351f7b6a3fa5

TS-11 - Unnecessary public function

ID	Type	Severity	Location	Status
TS-11	Gas Optimization	Informational	contracts/dao/DAOCommittee_V1.sol#L544 contracts/dao/DAOCommittee_V1.sol#L796 contracts/layer2/L1BridgeRegistryV1_1.sol#L340 contracts/layer2/L1BridgeRegistryV1_1.sol#L359 contracts/layer2/Layer2ManagerV1_1.sol#L357 contracts/layer2/Layer2ManagerV1_1.sol#L369 contracts/layer2/OperatorManagerV1_1.sol#L255 contracts/layer2/OperatorManagerV1_1.sol#L272	Resolved

Description

The mentioned functions are declared public and are meant to be callable both inside and outside the contract, but are not used internally.

Recommendation

Functions that are not used in the contract internal logic can be declared external to reduce gas consumption.

Alleviations

Fixed in the commit: b1c3e097babde0ea003ee8ec78b192a3304cc61e

TS-12 - Unused internal functions

ID	Type	Severity	Location	Status
TS-12	Gas Optimization	Informational	contracts/dao/DAOCommittee_V1.sol#L591 contracts/dao/DAOCommittee_V1.sol#L696	Resolved

Description

The internal functions mentioned are defined but are not executed from internal logic.

Recommendation

The team can reduce gas consumption when deploying contracts by removing unused functions.

Alleviations

Fixed in the commit: 4716824e7e1f63827bb9e697c309acb811c8b50a

TS-14 - Duplicate code (StorageStateCommittee)

ID	Type	Severity	Location	Status
TS-14	Gas Optimization	Informational	contracts/dao/StorageStateCommittee.sol#L13 contracts/dao/StorageStateCommittee.sol#L14	Resolved

Description

Defines AgendaStatus, AgendaResultenum types already implemented in the LibAgenda library, which not only reduces contract readability but also unnecessarily increases contract management points in subsequent LibAgenda library updates.

Recommendation

Delete AgendaStatus, AgendaResult types defined in the StorageStateCommittee contract.

Alleviations

Fixed in the commit: 0038fd0cbfa1585379f458d73eef1d822d884b27

TS-15 - Validate user input (CandidateAddOnV1_1.initialize)

ID	Type	Severity	Location	Status
TS-15	Volatile Code	Informational	contracts/dao/CandidateAddOnV1_1.sol#L46 contracts/dao/CandidateAddOnV1_1.sol#L47	Acknowledged

Description

The parameters of the initialize function, _ton, _wton, are directly entered by the user, but do not verify these values. If the team set a zero address in a state variable, the contract logic may not perform correctly in the future.

Recommendation

Perform verification on both parameters (_ton, _wton).

Alleviations

The development team said that the initialization of the CandiateAddOnV1_1 contract is no problem because CandiateAddOnFactory contract do the initialization.

TS-16 - Validate user input (address)

ID	Type	Severity	Location	Status
TS-16	Volatile Code	Informational	contracts/layer2/L1BridgeRegistryV1_1.sol#L153 contracts/layer2/Layer2ManagerV1_1.sol#L171 contracts/layer2/Layer2ManagerV1_1.sol#L186	Resolved

Description

The function parameters of the address type mentioned are directly entered by the user but do not verify this value. If zero address is stored in the state variable, subsequent contract logic may not be performed correctly.

Recommendation

Perform verification of address type parameter.

Alleviations

The development team said that for the function of stopping the issuance of L2 Signiorage, zero address is allowed in the seigniorageCommittee state variable of the L1BridgeRegistryV1_1 contract.

Other codes have been fixed in the commit: 9f297f070b6f70a1c967f032fb53f2fbf9260d41

TS-17 - Duplicate code (unstakedSeig)

ID	Type	Severity	Location	Status
TS-17	Gas Optimization	Informational	contracts/stake/managers/SeigManagerV1_3.sol#L622 contracts/stake/managers/SeigManagerV1_3.sol#L630	Resolved

Description

The totalPseig variable uses maxSeig − stakedSeig − l2TotalSeigs as the first factor in rmul, but this value performs the same operation when defining the unstakedSeig variable later.

```
uint256 totalPseig = rmul(maxSeig - stakedSeig - l2TotalSeigs, relativeSeigRate);
// ...
uint256 unstakedSeig = maxSeig - stakedSeig - l2TotalSeigs;
```

Running the same operation once can reduce gas consumption and increase code readability by avoiding the use of mathematical operators in the computation process of the totalPseig variable.

Recommendation

Please revise the code as below.

```
uint256 unstakedSeig = maxSeig - stakedSeig - l2TotalSeigs;
uint256 totalPseig = rmul(unstakedSeig, relativeSeigRate);
// ...
```

Alleviations

Fixed in the commit: 0824263859512899221f8103dbaa866ec678583f

TS-18 - Event emitted before state change

ID	Type	Severity	Location	Status
TS-18	Coding Style	Informational	contracts/layer2/OperatorManagerV1_1.sol#L154 contracts/stake/managers/SeigManagerV1_3.sol#L198	Resolved

Description

The mentioned codes are emiting an event before modifying the contract's state variable, which does not have a problem with the contract logic, but does not follow one of the Solidity patterns, Checks-Effects-Interactions, which makes it less readable.

Recommendation

If it do not necessarily have to follow the pattern, but the current implementation is not intended, make sure to modify the contract's state variable before generating the event.

Alleviations

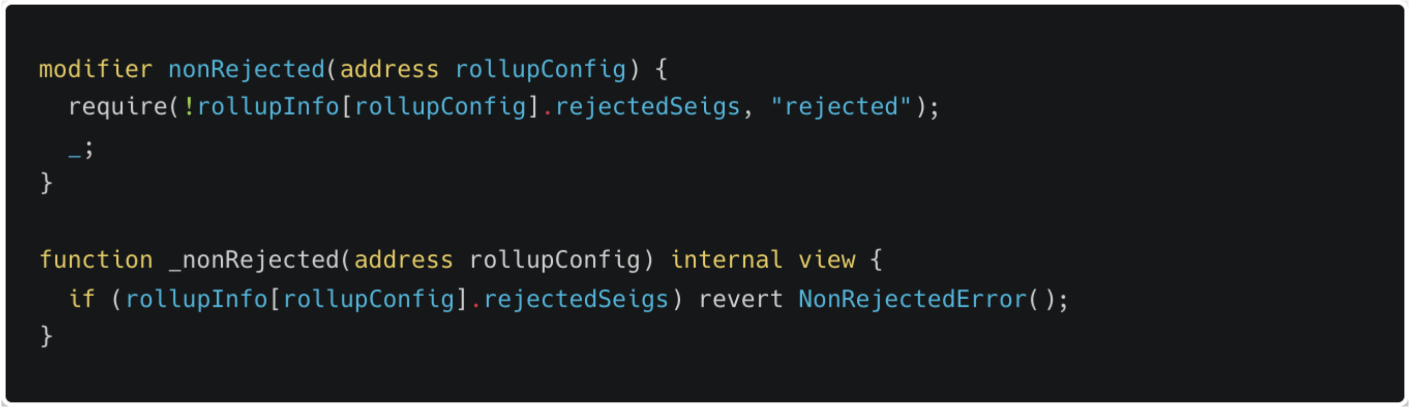
Fixed in the commit: 024ee55a80d44c3c2b6e04dd5e1157ba69291c5d

TS-19 - Duplicate code in modifier (nonRejected)

ID	Type	Severity	Location	Status
TS-19	Gas Optimization	Informational	contracts/layer2/L1BridgeRegistryV1_1.sol#L116	Resolved

Description

The require statement used inside the nonRejected modifier uses the same validation check as the _nonRejected function. The _nonRejected function is internal and uses custom error, so the team can use it instead of the require statement to reduce gas consumption.



```
modifier nonRejected(address rollupConfig) {
  require(!rollupInfo[rollupConfig].rejectedSeigs, "rejected");
  _;
}

function _nonRejected(address rollupConfig) internal view {
  if (rollupInfo[rollupConfig].rejectedSeigs) revert NonRejectedError();
}
```

Recommendation

Replace the require statement inside the nonRejected modifier with the _nonRejected function.

Alleviations

Fixed in the commit: 9e6c1d9016cb7e2b7aa3f8d213b08fbc034e4582

TS-20 - Typo (TOON)

ID	Type	Severity	Location	Status
TS-20	Coding Style	Informational	contracts/stake/managers/DepositManagerV1_1.sol#L78	Resolved

Description

The SELECTOR_SWAP_TOON_AND_TRANSFER variable is a variable that represents the 4 bytes signature of the swapToTONAndTransfer function. However, the variable name has a typo of TOON.

Recommendation

Please rename the variable SELECTOR_SWAP_TON_AND_TRANSFER or SELECTOR_SWAP_TO_TON_AND_TRANSFER.

Alleviations

Fixed in the commit: 12469a7c4ee17731ae43678fd6821614964707eb

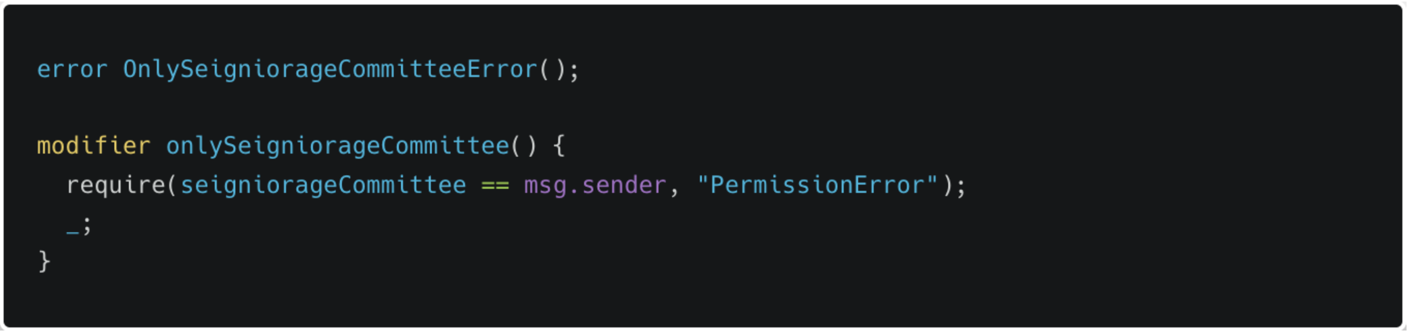
TS-21 - Use reason string instead of predefined error (onlySeigniorageCommittee)

ID	Type	Severity	Location	Status
TS-21	Gas Optimization	Informational	contracts/layer2/L1BridgeRegistryV1_1.sol#L26 contracts/layer2/L1BridgeRegistryV1_1.sol#L111	Resolved

Description

The onlySeigniorageCommittee modifier uses a requirement statement to verify the msg.sender's permissions. To replace this, OnlySeigniorageCommitteeError is declared, but it is not used by the onlySeigniorageCommittee modifier.

The requirement statement uses string literal, which increases gas consumption when deploying contracts. In addition, unnecessary definition of unused errors increases gas consumption.



```
error OnlySeigniorageCommitteeError();

modifier onlySeigniorageCommittee() {
  require(seigniorageCommittee == msg.sender, "PermissionError");
  _;
}
```

Recommendation

Use OnlySeigniorageCommitteeError inside the onlySeigniorageCommittee modifier to revert a transaction

Alleviations

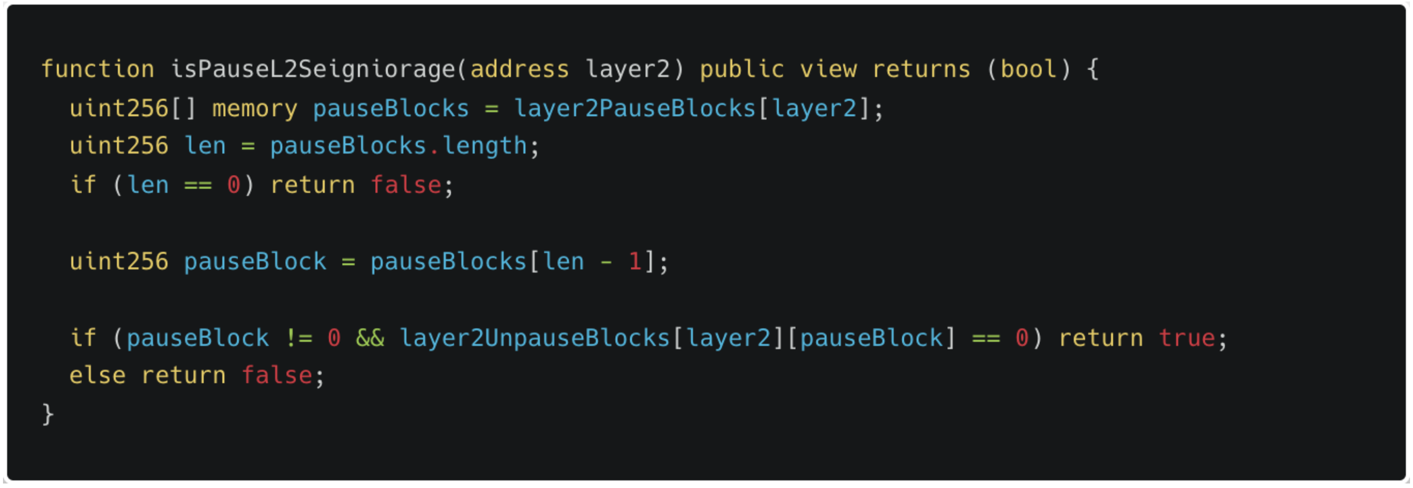
Fixed in the commit: 72d4f5ce6940537ddd9f054c579214b81454a4ce

TS-22 - Infinite length of array in memory

ID	Type	Severity	Location	Status
TS-22	Gas Optimization	Medium	contracts/stake/managers/SeigManagerV1_3.sol#L375	Resolved

Description

isPauseL2SeigniStorage function internally assigns an array of arbitrary lengths to the memory (pauseBlocks). The gas used by this assignment is proportional to the length of the array, which is likely to result in an out-of-gas error.



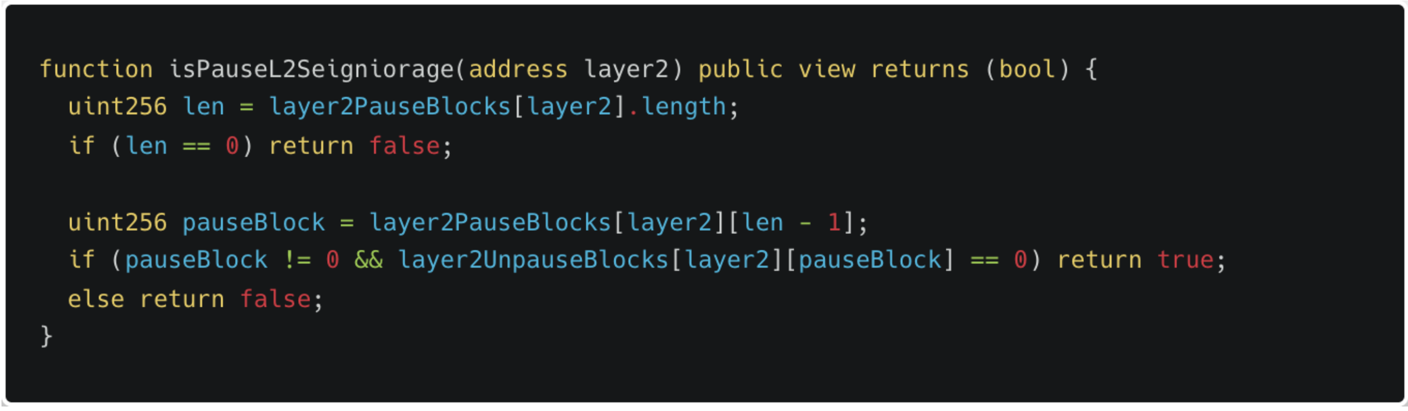
```
function isPauseL2Seigniorage(address layer2) public view returns (bool) {
  uint256[] memory pauseBlocks = layer2PauseBlocks[layer2];
  uint256 len = pauseBlocks.length;
  if (len == 0) return false;

  uint256 pauseBlock = pauseBlocks[len - 1];

  if (pauseBlock != 0 && layer2UnpauseBlocks[layer2][pauseBlock] == 0) return true;
  else return false;
}
```

Recommendation

Delete the pauseBlocks local variable.



```
function isPauseL2Seigniorage(address layer2) public view returns (bool) {
  uint256 len = layer2PauseBlocks[layer2].length;
  if (len == 0) return false;

  uint256 pauseBlock = layer2PauseBlocks[layer2][len - 1];
  if (pauseBlock != 0 && layer2UnpauseBlocks[layer2][pauseBlock] == 0) return true;
  else return false;
}
```

Alleviations

Fixed in the commit: 8e1928d59bce70287d227a6243bb7579ba74b9f3

TS-23 - Unused variable (_isSenderOperator)

ID	Type	Severity	Location	Status
TS-23	Gas Optimization	Informational	contracts/stake/managers/SeigManagerV1_3.sol#L543	Resolved

Description

The parameter(_isSenderOperator) of _increaseTot function is not used.

Recommendation

The team can reduce unnecessary contract deployment costs and increase code readability by deleting unused variables.

Alleviations

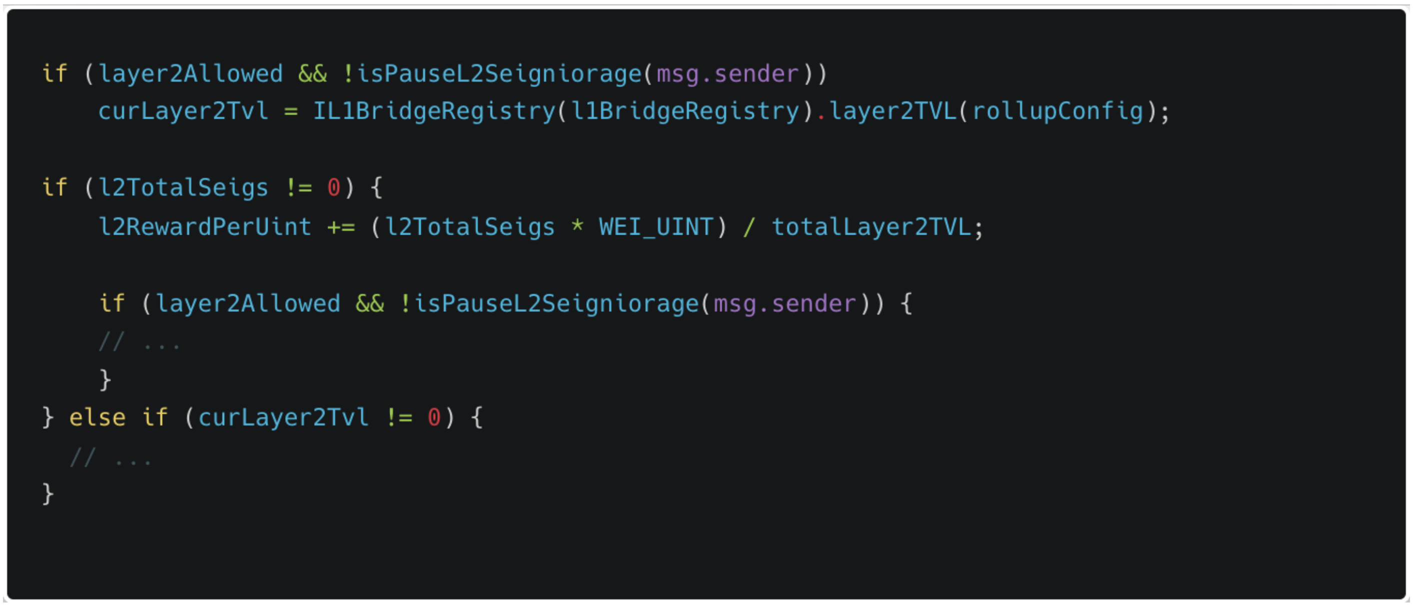
Fixed in the commit: ed2155e70d9a37d484767c3f1d3d4392445b3cbe

TS-24 - Duplicate code (isPauseL2Seigniorage(msg.sender))

ID	Type	Severity	Location	Status
TS-24	Gas Optimization	Informational	contracts/stake/managers/SeigManagerV1_3.sol#L624 contracts/stake/managers/SeigManagerV1_3.sol#L630	Resolved

Description

The same conditions (layer2Allowed &&!isPauseL2SeigniStorage (msg.sender) are evaluated twice, which reduces code readability and executes isPauseL2SeigniStorage twice to consume unnecessary gas.



```
if (layer2Allowed && !isPauseL2Seigniorage(msg.sender))
    curLayer2Tvl = IL1BridgeRegistry(l1BridgeRegistry).layer2TVL(rollupConfig);

if (l2TotalSeigs != 0) {
    l2RewardPerUint += (l2TotalSeigs * WEI_UINT) / totalLayer2TVL;

    if (layer2Allowed && !isPauseL2Seigniorage(msg.sender)) {
    // ...
    }
} else if (curLayer2Tvl != 0) {
  // ...
}
```

Recommendation

The team can assign those conditions to one local variable or adjust the order of the conditional statements to reduce gas consumption.

Alleviations

Fixed in the commit: c2a2c9a40e7cb3c116993154058079e041a9bb91

Finding Categories

Gas Optimization

Although it behaves logically the same as existing source code, EVM opcodes can be optimized to reduce transaction fees when deploying or executing a contract.

Control Flow

Consider whether it is correct for the contract to limit its function. This includes features that can only be performed by a specific user, such as an owner of the contract.

Volatile Code

Consider a particular edge case where a particular code behaves unexpectedly or a vulnerability exists.

Language Specific

Consider issues that occur only in Solidity.

Coding Style

The generated byte-code is not affected, but consider modifying the codebase to allow easily maintenance.