CAIRO SECURITY CLAN

SNAPSHOT X

SECURITY ASSESMENT REPORT

JUNE 2025

Prepared for

SNAPSHOT LABS

Contents

1 About Cairo Security Clan

Cairo Security Clan is a leading force in the realm of blockchain security, dedicated to fortifying the foundations of the digital age. As pioneers in the field, we specialize in conducting meticulous smart contract security audits, ensuring the integrity and reliability of decentralized applications built on blockchain technology.

At Cairo Security Clan, we boast a multidisciplinary team of seasoned professionals proficient in blockchain security, cryptography, and software engineering. With a firm commitment to excellence, our experts delve into every aspect of the Web3 ecosystem, from foundational layer protocols to application-layer development. Our comprehensive suite of services encompasses smart contract audits, formal verification, and real-time monitoring, offering unparalleled protection against potential vulnerabilities.

Our team comprises industry veterans and scholars with extensive academic backgrounds and practical experience. Armed with advanced methodologies and cutting-edge tools, we scrutinize and analyze complex smart contracts with precision and rigor. Our track record speaks volumes, with a plethora of published research papers and citations, demonstrating our unwavering dedication to advancing the field of blockchain security.

At Cairo Security Clan, we prioritize collaboration and transparency, fostering meaningful partnerships with our clients. We believe in a customer-oriented approach, engaging stakeholders at every stage of the auditing process. By maintaining open lines of communication and soliciting client feedback, we ensure that our solutions are tailored to meet the unique needs and objectives of each project.

Beyond our core services, Cairo Security Clan is committed to driving innovation and shaping the future of blockchain technology. As active contributors to the ecosystem, we participate in the development of emerging technologies such as Starknet, leveraging our expertise to build robust infrastructure and tools. Through strategic guidance and support, we empower our partners to navigate the complexities of the blockchain landscape with confidence and clarity.

In summary, Cairo Security Clan stands at the forefront of blockchain security, blending technical prowess with a client-centric ethos to deliver unparalleled protection and peace of mind in an ever-evolving digital landscape. Join us in safeguarding the future of decentralized finance and digital assets with confidence and conviction.

2 Disclaimer

Disclaimer Limitations of this Audit:

This report is based solely on the materials and documentation provided by you to Cairo Security Clan for the specific purpose of conducting the security review outlined in the Summary of Audit and Scoped Files. The findings presented here may not be exhaustive and may not identify all potential vulnerabilities. Cairo Security Clan provides this review and report on an "as-is" and "as-available" basis. You acknowledge that your use of this report, including any associated services, products, protocols, platforms, content, and materials, occurs entirely at your own risk.

Inherent Risks of Blockchain Technology:

Blockchain technology remains in its developmental stage and is inherently susceptible to unknown risks and vulnerabilities. This review is specifically focused on the smart contract code and does not extend to the compiler layer, programming language elements beyond the reviewed code, or other potential security risks outside the code itself.

Report Purpose and Reliance:

This report should not be construed as an endorsement of any specific project or team, nor does it guarantee the absolute security of the audited smart contracts. No third party should rely on this report for any purpose, including making investment or purchasing decisions.

Liability Disclaimer:

To the fullest extent permitted by law, Cairo Security Clan disclaims all liability associated with this report, its contents, and any related services and products arising from your use. This includes, but is not limited to, implied warranties of merchantability, fitness for a particular purpose, and non-infringement.

Third-Party Products and Services:

Cairo Security Clan does not warrant, endorse, guarantee, or assume responsibility for any products or services advertised by third parties within this report, nor for any open-source or third-party software, code, libraries, materials, or information linked to, referenced by, or accessible through this report, its content, and related services and products. This includes any hyperlinked websites, websites or applications appearing on advertisements, and Cairo Security Clan will not be responsible for monitoring any transactions between you and third-party providers. It is recommended that you exercise due diligence and caution when considering any third-party products or services, just as you would with any purchase or service through any medium.

Disclaimer of Advice:

FOR THE AVOIDANCE OF DOUBT, THIS REPORT, ITS CONTENT, ACCESS, AND/OR USE, INCLUDING ANY ASSOCIATED SERVICES OR MATERIALS, SHOULD NOT BE CONSIDERED OR RELIED UPON AS FINANCIAL, INVESTMENT, TAX, LEGAL, REGULATORY, OR OTHER PROFESSIONAL ADVICE.

3 Executive Summary

This document presents the security review performed by Cairo Security Clan on the Starknet implementation of Snapshop X.

Snapshot is an off-chain, gasless voting platform designed for DAOs, DeFi, and NFT communities, enabling customizable voting processes. It offers flexible voting strategies, multiple voting systems, and robust proposal validation. Votes are submitted as signed messages, ensuring easy verification and integrity of results. Learn more from docs.

Our audit engagement specifically focuses on the new code updates and differences introduced. These are:

- `MerkleWhitelistVotingStrategy.sol`: Introduces a new voting strategy module that uses Merkle trees to efficiently define and verify voting power and voter eligibility.
- `DelegateRegistry.sol`: Manages delegation relationships between delegators and delegates. It allows users to set or clear delegates, tracks delegation counts, and provides reverse lookup functionality for external applications.
- `ApeGasVotingStrategy.sol`: Introduces a new voting strategy module that calculates voting power based on a user's Ape Gas balance. Utilizes Herodotus storage proofs infrastructure to compute voting power at a specific block number, mapping L1 block number to L3 via timestamp correlation.
- `SignatureVerifier.sol`: The contract has been updated to support EIP-1271, which now enables smart contracts to validate signatures. This enhancement is implemented through OpenZeppelin's SignatureChecker, replacing the previous ECDSA-only verification approach.
- `execute()` function update: The function has been updated across all execution strategies to include proposalId as one of the parameters, though currently it remains unused.

The audit was performed using

- manual analysis of the codebase,
- automated analysis tools,
- simulation of the smart contract,
- analysis of edge test cases

This document is organized as follows. Section 1 About Cairo Security Clan. Section 2 Disclaimer. Section 3 Executive Summary. Section 4 Summary of Audit. Section 5 Risk Classification. Section 6 Issues by Severity Levels. Section 7 Test Evaluation.

No issues found in this security review.

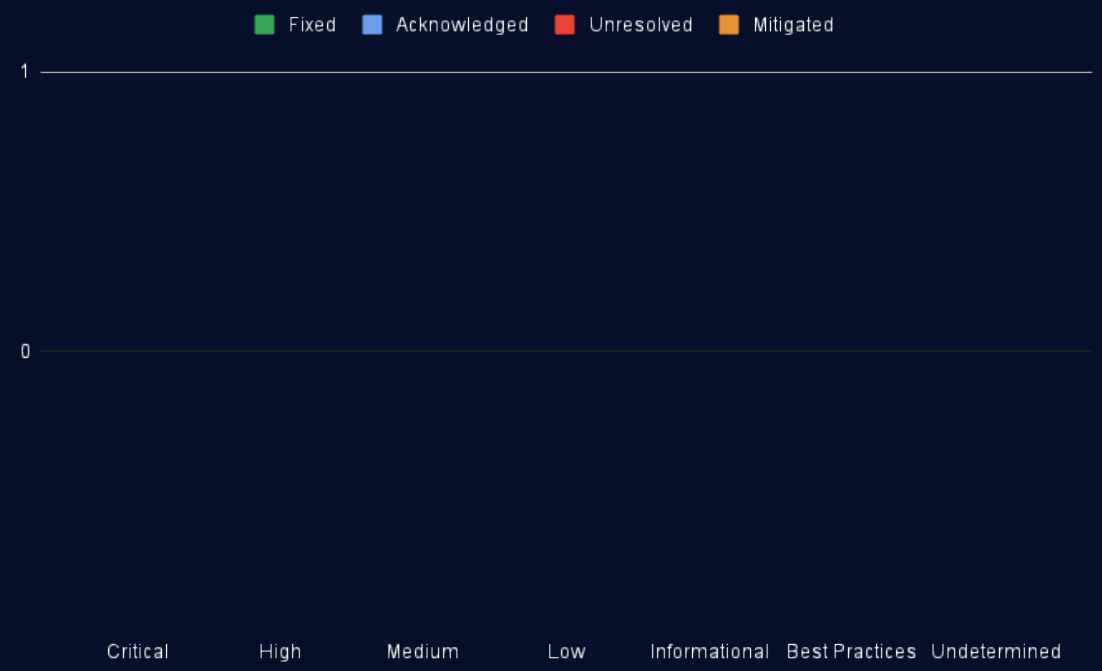

Fig 1: Distribution of issues: Critical (0), **High** (0), **Medium** (0), **Low** (0), **Informational** (0), **Best Practices** (0), **Undetermined** (0). **Distribution of status: Fixed** (0), **Acknowledged** (0), **Mitigated** (0), **Unresolved** (0).

4 Summary of Audit

Audit Type	Security Review of Difference Between Commits (Best Effort)
Solidity Version	0.8.18
Final Report	09/06/2025
Repository	snapshot-labs/sx-evm
Initial Commit Hash	6c7830f9466ad4f6324afa6b4e02fc3818791162
Difference Between Hashes	6c7830f...7804cb1
Documentation	Website documentation
Test Suite Assessment	High

4.1 Scoped Files

	Contracts
1	src/utils/SignatureVerifier.sol
2	src/voting-strategies/ApeGasVotingStrategy.sol
3	src/voting-strategies/MerkleWhitelistVotingStrategy.sol

4.2 Issues

No issues found in this security review.

5 Risk Classification

The risk rating methodology used by Cairo Security Clan follows the principles established by the CVSS risk rating methodology. The severity of each finding is determined by two factors: **Likelihood** and **Impact**.

Likelihood measures how likely an attacker will uncover and exploit the finding. This factor will be one of the following values:

a) **High**: The issue is trivial to exploit and has no specific conditions that need to be met;

b) **Medium**: The issue is moderately complex and may have some conditions that need to be met;

c) **Low**: The issue is very complex and requires very specific conditions to be met.

When defining the likelihood of a finding, other factors are also considered. These can include but are not limited to Motive, opportunity, exploit accessibility, ease of discovery, and ease of exploit.

Impact is a measure of the damage that may be caused if an attacker exploits the finding. This factor will be one of the following values:

a) **High**: The issue can cause significant damage such as loss of funds or the protocol entering an unrecoverable state;

b) **Medium**: The issue can cause moderate damage such as impacts that only affect a small group of users or only a particular part of the protocol;

c) **Low**: The issue can cause little to no damage such as bugs that are easily recoverable or cause unexpected interactions that cause minor inconveniences.

When defining the impact of a finding other factors are also considered. These can include but are not limited to Data/state integrity, loss of availability, financial loss, and reputation damage. After defining the likelihood and impact of an issue, the severity can be determined according to the table below.

		Likelihood		
		High	**Medium**	**Low**
Impact	**High**	Critical	High	Medium
	Medium	High	Medium	Low
	Low	Medium	Low	Info/Best Practices

To address issues that do not fit a High/Medium/Low severity, Cairo Security Clan also uses three more finding severities: **Informational**, **Best Practices** and **Gas**

a) **Informational** findings do not pose any risk to the application, but they carry some information that the audit team intends to formally pass to the client;

b) **Best Practice** findings are used when some piece of code does not conform with smart contract development best practices;

c) **Gas** findings are used when some piece of code uses more gas than it should be or have some functions that can be removed to save gas.

6 Issues by Severity Levels

No issues found in this security review.

7 Test & Compilation Evaluation

7.1 Compilation Output

```
forge build
Compiling...
Compiling 197 files with Solc 0.8.27
Solc 0.8.27 finished in 255.87s
Compiler run successful with warnings:
Warning (2018): Function state mutability can be restricted to view
  --> lib/forge-gas-snapshot/src/GasSnapshot.sol:82:5:
   |
82 |     function _readSnapshot(string memory name) private returns (uint256 res) {
   |     ^ (Relevant source part starts here and spans across multiple lines).
```

7.2 Tests Output

```
forge test
Compiling...
No files changed, compilation skipped

Ran 19 tests for test/AvatarExecutionStrategy.t.sol:AvatarExecutionStrategyTestDirect
[PASS] testDisableInvalidSpace() (gas: 16489)
[PASS] testDisableSpace() (gas: 274645)
[PASS] testDoubleInitialization() (gas: 16591)
[PASS] testEnableInvalidSpace() (gas: 13842)
[PASS] testEnableSpace() (gas: 39091)
[PASS] testEnableSpaceTwice() (gas: 17597)
[PASS] testExecution() (gas: 321586)
[PASS] testExecutionInvalidPayload() (gas: 256067)
[PASS] testExecutionProposalNotAccepted() (gas: 199136)
[PASS] testGetStrategyType() (gas: 10035)
[PASS] testInvalidMultiTx() (gas: 329417)
[PASS] testInvalidTx() (gas: 286398)
[PASS] testMultiTx() (gas: 350638)
[PASS] testSetTarget() (gas: 22982)
[PASS] testTransferOwnership() (gas: 19534)
[PASS] testUnauthorizedDisableSpace() (gas: 15782)
[PASS] testUnauthorizedEnableSpace() (gas: 14494)
[PASS] testUnauthorizedSetTarget() (gas: 14208)
[PASS] testUnauthorizedTransferOwnership() (gas: 13498)
Suite result: ok. 19 passed; 0 failed; 0 skipped; finished in 22.15ms (5.16ms CPU time)

Ran 19 tests for test/AvatarExecutionStrategy.t.sol:AvatarExecutionStrategyTestProxy
[PASS] testDisableInvalidSpace() (gas: 21300)
[PASS] testDisableSpace() (gas: 280175)
[PASS] testDoubleInitialization() (gas: 21447)
[PASS] testEnableInvalidSpace() (gas: 18653)
[PASS] testEnableSpace() (gas: 44208)
[PASS] testEnableSpaceTwice() (gas: 22408)
[PASS] testExecution() (gas: 326890)
[PASS] testExecutionInvalidPayload() (gas: 256067)
[PASS] testExecutionProposalNotAccepted() (gas: 204080)
[PASS] testGetStrategyType() (gas: 14848)
[PASS] testInvalidMultiTx() (gas: 334413)
[PASS] testInvalidTx() (gas: 291339)
[PASS] testMultiTx() (gas: 355630)
[PASS] testSetTarget() (gas: 28096)
[PASS] testTransferOwnership() (gas: 24648)
[PASS] testUnauthorizedDisableSpace() (gas: 20605)
[PASS] testUnauthorizedEnableSpace() (gas: 19317)
[PASS] testUnauthorizedSetTarget() (gas: 19031)
[PASS] testUnauthorizedTransferOwnership() (gas: 18321)
Suite result: ok. 19 passed; 0 failed; 0 skipped; finished in 24.03ms (7.54ms CPU time)

Ran 14 tests for test/EmergencyQuorumExecutionStrategy.t.sol:EmergencyQuorumTest
```

```
[PASS] testEmergencyQuorum() (gas: 328846)
[PASS] testEmergencyQuorumAfterMaxDuration() (gas: 290554)
[PASS] testEmergencyQuorumAfterMinDuration() (gas: 290451)
[PASS] testEmergencyQuorumAlreadyExecuted() (gas: 293942)
[PASS] testEmergencyQuorumCancelled() (gas: 255184)
[PASS] testEmergencyQuorumLowerThanQuorum() (gas: 1119235)
[PASS] testEmergencyQuorumNotReached() (gas: 264183)
[PASS] testEmergencyQuorumReachedButRejected() (gas: 343517)
[PASS] testEmergencyQuorumSetEmergencyQuorum() (gas: 448039)
[PASS] testEmergencyQuorumSetEmergencyQuorumUnauthorized() (gas: 11375)
[PASS] testEmergencyQuorumSetQuorum() (gas: 379910)
[PASS] testEmergencyQuorumSetQuorumUnauthorized() (gas: 14200)
[PASS] testEmergencyQuorumVotingPeriod() (gas: 299436)
[PASS] testGetStrategyType() (gas: 9905)
Suite result: ok. 14 passed; 0 failed; 0 skipped; finished in 24.33ms (8.50ms CPU time)

Ran 4 tests for test/OZVotesVotingStrategy.t.sol:OZVotesVotingStrategyTest
[PASS] testGetVotingPower() (gas: 187008)
[PASS] testGetVotingPowerInvalidParamsArray() (gas: 184791)
[PASS] testGetVotingPowerInvalidToken() (gas: 185361)
[PASS] testGetZeroVotingPower() (gas: 117211)
Suite result: ok. 4 passed; 0 failed; 0 skipped; finished in 1.49ms (816.84s CPU time)

Ran 6 tests for test/ActiveProposalsLimiterProposalValidationStrategy.t.sol:ActiveProposalsLimterTest
[PASS] testDecreaseProposalLimit() (gas: 653700)
[PASS] testSpamMaxProposals() (gas: 616501)
[PASS] testSpamMinusOneProposals() (gas: 603518)
[PASS] testSpamOneProposal() (gas: 173662)
[PASS] testSpamOverLongPeriod() (gas: 1150024)
[PASS] testSpamThenWaitThenSpam() (gas: 1164022)
Suite result: ok. 6 passed; 0 failed; 0 skipped; finished in 26.74ms (8.70ms CPU time)

Ran 9 tests for test/EthTxAuthenticator.t.sol:EthTxAuthenticatorTest
[PASS] testAuthenticateTxPropose() (gas: 159825)
[PASS] testAuthenticateTxProposeInvalidAuthor() (gas: 31986)
[PASS] testAuthenticateTxProposeInvalidSelector() (gas: 31076)
[PASS] testAuthenticateTxUpdateProposal() (gas: 369405)
[PASS] testAuthenticateTxUpdateProposalInvalidCaller() (gas: 21965)
[PASS] testAuthenticateTxUpdateProposalInvalidSelector() (gas: 16174)
[PASS] testAuthenticateTxVote() (gas: 230817)
[PASS] testAuthenticateTxVoteInvalidSelector() (gas: 23993)
[PASS] testAuthenticateTxVoteInvalidVoter() (gas: 25168)
Suite result: ok. 9 passed; 0 failed; 0 skipped; finished in 11.02ms (2.90ms CPU time)

Ran 22 tests for test/CompTimelockExecutionStrategy.t.sol:CompTimelockExecutionStrategyTestDirect
[PASS] testExecute() (gas: 356834)
[PASS] testExecuteBeforeDelay() (gas: 345777)
[PASS] testExecuteDelegateCall() (gas: 368829)
[PASS] testExecuteDoubleExecution() (gas: 358326)
[PASS] testExecuteInvalidPayload() (gas: 346812)
[PASS] testExecuteNFTs() (gas: 1452329)
[PASS] testExecuteNotQueued() (gas: 235082)
[PASS] testExecuteTransactionFailed() (gas: 364009)
[PASS] testQueueing() (gas: 341868)
[PASS] testQueueingDoubleQueue() (gas: 347556)
[PASS] testQueueingDuplicateMetaTransaction() (gas: 342543)
[PASS] testQueueingDuplicateMetaTransactionDifferentProposals() (gas: 544263)
[PASS] testQueueingFromUnauthorizedSpace() (gas: 269886)
[PASS] testQueueingInvalidPayload() (gas: 255300)
[PASS] testQueueingQueueDuplicate() (gas: 518477)
[PASS] testQueueingRejectedProposal() (gas: 200432)
[PASS] testSetUp() (gas: 1669174)
[PASS] testSetVetoGuardian() (gas: 18629)
[PASS] testVetoOnlyGuardian() (gas: 343756)
[PASS] testVetoProposal() (gas: 350541)
[PASS] testVetoProposalNotQueued() (gas: 261900)
[PASS] testViewFunctions() (gas: 9593)
Suite result: ok. 22 passed; 0 failed; 0 skipped; finished in 38.38ms (21.19ms CPU time)
```

```
Ran 22 tests for test/OptimisticCompTimelockExecutionStrategy.t.sol:
    OptimisticCompTimelockExecutionStrategyTestDirect
[PASS] testExecute() (gas: 356667)
[PASS] testExecuteBeforeDelay() (gas: 345610)
[PASS] testExecuteDelegateCall() (gas: 368662)
[PASS] testExecuteDoubleExecution() (gas: 358159)
[PASS] testExecuteInvalidPayload() (gas: 346645)
[PASS] testExecuteNFTs() (gas: 1452162)
[PASS] testExecuteNotQueued() (gas: 235082)
[PASS] testExecuteTransactionFailed() (gas: 363842)
[PASS] testQueueing() (gas: 271362)
[PASS] testQueueingDoubleQueue() (gas: 347389)
[PASS] testQueueingDuplicateMetaTransaction() (gas: 342376)
[PASS] testQueueingDuplicateMetaTransactionDifferentProposals() (gas: 488084)
[PASS] testQueueingFromUnauthorizedSpace() (gas: 269886)
[PASS] testQueueingInvalidPayload() (gas: 255300)
[PASS] testQueueingQueueDuplicate() (gas: 518174)
[PASS] testQueueingRejectedProposal() (gas: 270569)
[PASS] testSetUp() (gas: 1654532)
[PASS] testSetVetoGuardian() (gas: 18629)
[PASS] testVetoOnlyGuardian() (gas: 343589)
[PASS] testVetoProposal() (gas: 350374)
[PASS] testVetoProposalNotQueued() (gas: 261900)
[PASS] testViewFunctions() (gas: 9593)
Suite result: ok. 22 passed; 0 failed; 0 skipped; finished in 41.48ms (20.54ms CPU time)

Ran 1 test for test/GasSnapshots.t.sol:GasSnapshotsTest
[PASS] testVoteAndProposeWithCompToken() (gas: 792305)
Suite result: ok. 1 passed; 0 failed; 0 skipped; finished in 20.76ms (11.50ms CPU time)

Ran 13 tests for test/EthSigAuthenticator.t.sol:EthSigAuthenticatorTest
[PASS] testAuthenticateInvalidSignature() (gas: 70320)
[PASS] testAuthenticatePropose() (gas: 194173)
[PASS] testAuthenticateProposeInvalidSelector() (gas: 37817)
[PASS] testAuthenticateProposeInvalidSigner() (gas: 69809)
[PASS] testAuthenticateProposeReusedSignature() (gas: 201008)
[PASS] testAuthenticateUpdateProposal() (gas: 267189)
[PASS] testAuthenticateUpdateProposalInvalidSelector() (gas: 29941)
[PASS] testAuthenticateUpdateProposalInvalidSignature() (gas: 256846)
[PASS] testAuthenticateVote() (gas: 246903)
[PASS] testAuthenticateVoteInvalidSelector() (gas: 33256)
[PASS] testAuthenticateVoteInvalidSignature() (gas: 44151)
[PASS] testAuthenticateVoteInvalidSigner() (gas: 43790)
[PASS] testAuthenticateVoteReusedSignature() (gas: 262519)
Suite result: ok. 13 passed; 0 failed; 0 skipped; finished in 25.10ms (17.40ms CPU time)

Ran 1 test for test/VanillaVotingStrategy.t.sol:VanillaVotingStrategyTest
[PASS] testGetVotingPower() (gas: 8941)
Suite result: ok. 1 passed; 0 failed; 0 skipped; finished in 297.48s (51.68s CPU time)

Ran 11 tests for test/Execute.t.sol:ExecuteTest
[PASS] testExecute() (gas: 292522)
[PASS] testExecuteAlreadyExecuted() (gas: 290151)
[PASS] testExecuteInvalidExecutionStrategy() (gas: 254923)
[PASS] testExecuteInvalidPayload() (gas: 252607)
[PASS] testExecuteInvalidProposal() (gas: 257900)
[PASS] testExecuteMinDurationNotElapsed() (gas: 360160)
[PASS] testExecuteQuorumNotReachedAtAll() (gas: 207762)
[PASS] testExecuteQuorumNotReachedYet() (gas: 190517)
[PASS] testExecuteWithAbstainVote() (gas: 276349)
[PASS] testExecuteWithAgainstVote() (gas: 276453)
[PASS] testGetStrategyType() (gas: 9883)
Suite result: ok. 11 passed; 0 failed; 0 skipped; finished in 14.58ms (11.82ms CPU time)

Ran 13 tests for test/Vote.t.sol:VoteTest
[PASS] testVote() (gas: 226542)
[PASS] testVoteAddedVotingStrategy() (gas: 256750)
[PASS] testVoteAlreadyExecuted() (gas: 291744)
[PASS] testVoteDoubleVote() (gas: 230344)
[PASS] testVoteDuplicateUsedVotingStrategy() (gas: 191295)
```

```
[PASS] testVoteInvalidAuth() (gas: 163354)
[PASS] testVoteInvalidProposalId() (gas: 173219)
[PASS] testVoteInvalidVotingStrategy() (gas: 181998)
[PASS] testVoteMultipleStrategies() (gas: 497808)
[PASS] testVoteNoVotingPower() (gas: 180436)
[PASS] testVoteRemovedVotingStrategy() (gas: 292208)
[PASS] testVoteVotingPeriodHasEnded() (gas: 167323)
[PASS] testVoteVotingPeriodHasNotStarted() (gas: 285303)
Suite result: ok. 13 passed; 0 failed; 0 skipped; finished in 15.83ms (8.98ms CPU time)

Ran 22 tests for test/OptimisticCompTimelockExecutionStrategy.t.sol:
     OptimisticCompTimelockExecutionStrategyTestProxy
[PASS] testExecute() (gas: 362275)
[PASS] testExecuteBeforeDelay() (gas: 350915)
[PASS] testExecuteDelegateCall() (gas: 373621)
[PASS] testExecuteDoubleExecution() (gas: 364135)
[PASS] testExecuteInvalidPayload() (gas: 352257)
[PASS] testExecuteNFTs() (gas: 1457904)
[PASS] testExecuteNotQueued() (gas: 239950)
[PASS] testExecuteTransactionFailed() (gas: 389366)
[PASS] testQueueing() (gas: 276299)
[PASS] testQueueingDoubleQueue() (gas: 352326)
[PASS] testQueueingDuplicateMetaTransaction() (gas: 347359)
[PASS] testQueueingDuplicateMetaTransactionDifferentProposals() (gas: 493462)
[PASS] testQueueingFromUnauthorizedSpace() (gas: 275134)
[PASS] testQueueingInvalidPayload() (gas: 255300)
[PASS] testQueueingQueueDuplicate() (gas: 523552)
[PASS] testQueueingRejectedProposal() (gas: 275513)
[PASS] testSetUp() (gas: 1654448)
[PASS] testSetVetoGuardian() (gas: 23759)
[PASS] testVetoOnlyGuardian() (gas: 348894)
[PASS] testVetoProposal() (gas: 356657)
[PASS] testVetoProposalNotQueued() (gas: 267075)
[PASS] testViewFunctions() (gas: 14412)
Suite result: ok. 22 passed; 0 failed; 0 skipped; finished in 24.62ms (17.53ms CPU time)

Ran 22 tests for test/CompTimelockExecutionStrategy.t.sol:CompTimelockExecutionStrategyTestProxy
[PASS] testExecute() (gas: 362442)
[PASS] testExecuteBeforeDelay() (gas: 351082)
[PASS] testExecuteDelegateCall() (gas: 373788)
[PASS] testExecuteDoubleExecution() (gas: 364302)
[PASS] testExecuteInvalidPayload() (gas: 352424)
[PASS] testExecuteNFTs() (gas: 1458071)
[PASS] testExecuteNotQueued() (gas: 239950)
[PASS] testExecuteTransactionFailed() (gas: 389533)
[PASS] testQueueing() (gas: 346805)
[PASS] testQueueingDoubleQueue() (gas: 352493)
[PASS] testQueueingDuplicateMetaTransaction() (gas: 347526)
[PASS] testQueueingDuplicateMetaTransactionDifferentProposals() (gas: 549641)
[PASS] testQueueingFromUnauthorizedSpace() (gas: 275134)
[PASS] testQueueingInvalidPayload() (gas: 255300)
[PASS] testQueueingQueueDuplicate() (gas: 523855)
[PASS] testQueueingRejectedProposal() (gas: 205376)
[PASS] testSetUp() (gas: 1669090)
[PASS] testSetVetoGuardian() (gas: 23759)
[PASS] testVetoOnlyGuardian() (gas: 349061)
[PASS] testVetoProposal() (gas: 356824)
[PASS] testVetoProposalNotQueued() (gas: 267075)
[PASS] testViewFunctions() (gas: 14412)
Suite result: ok. 22 passed; 0 failed; 0 skipped; finished in 28.50ms (22.56ms CPU time)

Ran 6 tests for test/ProxyFactory.t.sol:SpaceFactoryTest
[PASS] testCreateSpace() (gas: 554984)
[PASS] testCreateSpaceFailedInitialization() (gas: 181458)
[PASS] testCreateSpaceInvalidImplementation() (gas: 63384)
[PASS] testCreateSpaceReInitialize() (gas: 564458)
[PASS] testCreateSpaceReusedSalt() (gas: 555501)
[PASS] testPredictProxyAddress() (gas: 14545)
Suite result: ok. 6 passed; 0 failed; 0 skipped; finished in 9.14ms (6.44ms CPU time)
```

```
Ran 4 tests for test/CompVotingStrategy.t.sol:CompVotingStrategyTest
[PASS] testGetVotingPower() (gas: 187253)
[PASS] testGetVotingPowerInvalidParamsArray() (gas: 184794)
[PASS] testGetVotingPowerInvalidToken() (gas: 185386)
[PASS] testGetZeroVotingPower() (gas: 117486)
Suite result: ok. 4 passed; 0 failed; 0 skipped; finished in 1.79ms (985.40s CPU time)

Ran 2 tests for test/SimpleQuorum.t.sol:SimpleQuorumTest
[PASS] test_SimpleQuorumSetQuorum() (gas: 374279)
[PASS] test_SimpleQuorumSetQuorumUnauthorized() (gas: 14087)
Suite result: ok. 2 passed; 0 failed; 0 skipped; finished in 6.78ms (879.56s CPU time)

Ran 3 tests for test/WhitelistVotingStrategy.t.sol:WhitelistVotingStrategyTest
[PASS] testWhitelistIndexOutOfBounds() (gas: 205981)
[PASS] testWhitelistVoterAndIndexMismatch() (gas: 206287)
[PASS] testWhitelistVotingPower() (gas: 213183)
Suite result: ok. 3 passed; 0 failed; 0 skipped; finished in 1.26ms (662.68s CPU time)

Ran 5 tests for test/VotingPowerProposalValidationStrategy.t.sol:PropositionPowerProposalValidationTest
[PASS] testPropose() (gas: 186464)
[PASS] testProposeDuplicateUserVotingStrategy() (gas: 78059)
[PASS] testProposeInsufficientVotingPower() (gas: 171054)
[PASS] testProposeInvalidUserVotingStrategy() (gas: 72495)
[PASS] testProposeMultipleVotingStrategies() (gas: 747480)
Suite result: ok. 5 passed; 0 failed; 0 skipped; finished in 12.09ms (2.14ms CPU time)

Ran 1 test for test/Deployer.t.sol:DeployerTest
[PASS] testDeployer() (gas: 15958092)
Suite result: ok. 1 passed; 0 failed; 0 skipped; finished in 12.58ms (8.32ms CPU time)

Ran 3 tests for test/Propose.t.sol:ProposeTest
[PASS] testPropose() (gas: 164024)
[PASS] testProposeInvalidAuth() (gas: 33232)
[PASS] testProposeRefusedValidation() (gas: 156851)
Suite result: ok. 3 passed; 0 failed; 0 skipped; finished in 8.25ms (974.24s CPU time)

Ran 2 tests for test/VanillaAuthenticator.t.sol:VanillaAuthenticatorTest
[PASS] testAuthenticate() (gas: 37162)
[PASS] testAuthenticateRevert() (gas: 14428)
Suite result: ok. 2 passed; 0 failed; 0 skipped; finished in 452.83s (160.88s CPU time)

Ran 22 tests for test/OptimisticTimelockExecutionStrategy.t.sol:OptimisticTimelockExecutionStrategyTestProxy
[PASS] testCheckViewFunctions() (gas: 18883)
[PASS] testExecute() (gas: 336715)
[PASS] testExecuteBeforeDelay() (gas: 313051)
[PASS] testExecuteDelegateCall() (gas: 406936)
[PASS] testExecuteDoubleExecution() (gas: 338546)
[PASS] testExecuteInvalidPayload() (gas: 313700)
[PASS] testExecuteNFTs() (gas: 2656138)
[PASS] testExecuteNotQueued() (gas: 239879)
[PASS] testExecuteTransactionFailed() (gas: 347873)
[PASS] testQueueing() (gas: 238494)
[PASS] testQueueingDoubleQueue() (gas: 314220)
[PASS] testQueueingFromUnauthorizedSpace() (gas: 275108)
[PASS] testQueueingInvalidPayload() (gas: 255252)
[PASS] testQueueingQueueDuplicate() (gas: 485660)
[PASS] testQueueingQueueDuplicateUniqueSalt() (gas: 515071)
[PASS] testQueueingRejectedProposal() (gas: 275132)
[PASS] testSetTimelockDelay() (gas: 23840)
[PASS] testSetTimelockDelayUnauthorized() (gas: 18907)
[PASS] testSetUp() (gas: 38718)
[PASS] testVetoOnlyGuardian() (gas: 310748)
[PASS] testVetoProposal() (gas: 326700)
[PASS] testVetoUnqueuedProposal() (gas: 266889)
Suite result: ok. 22 passed; 0 failed; 0 skipped; finished in 42.51ms (22.82ms CPU time)

Ran 5 tests for test/UpdateProposal.t.sol:UpdateProposalTest
[PASS] testUpdateFinalizedProposal() (gas: 299128)
[PASS] testUpdateProposal() (gas: 306594)
[PASS] testUpdateProposalAfterDelay() (gas: 165501)
```

```
[PASS] testUpdateProposalInvalidCaller() (gas: 161638)
[PASS] testUpdateProposalUnauthenticated() (gas: 160975)
Suite result: ok. 5 passed; 0 failed; 0 skipped; finished in 13.79ms (1.99ms CPU time)

Ran 22 tests for test/TimelockExecutionStrategy.t.sol:TimelockExecutionStrategyTestProxy
[PASS] testCheckViewFunctions() (gas: 18889)
[PASS] testExecute() (gas: 336894)
[PASS] testExecuteBeforeDelay() (gas: 313230)
[PASS] testExecuteDelegateCall() (gas: 407115)
[PASS] testExecuteDoubleExecution() (gas: 338725)
[PASS] testExecuteInvalidPayload() (gas: 313878)
[PASS] testExecuteNFTs() (gas: 2656316)
[PASS] testExecuteNotQueued() (gas: 239891)
[PASS] testExecuteTransactionFailed() (gas: 348051)
[PASS] testQueueing() (gas: 309005)
[PASS] testQueueingDoubleQueue() (gas: 314398)
[PASS] testQueueingFailedProposal() (gas: 203663)
[PASS] testQueueingFromUnauthorizedSpace() (gas: 275118)
[PASS] testQueueingInvalidPayload() (gas: 255262)
[PASS] testQueueingQueueDuplicate() (gas: 485974)
[PASS] testQueueingQueueDuplicateUniqueSalt() (gas: 515408)
[PASS] testSetTimelockDelay() (gas: 23851)
[PASS] testSetTimelockDelayUnauthorized() (gas: 18918)
[PASS] testSetUp() (gas: 38729)
[PASS] testVetoOnlyGuardian() (gas: 310925)
[PASS] testVetoProposal() (gas: 326878)
[PASS] testVetoUnqueuedProposal() (gas: 266900)
Suite result: ok. 22 passed; 0 failed; 0 skipped; finished in 15.76ms (18.34ms CPU time)

Ran 30 tests for test/SpaceOwnerActions.t.sol:SpaceOwnerActionsTest
[PASS] testAddAndRemoveAuthenticators() (gas: 207668)
[PASS] testAddAndRemoveVotingStrategies() (gas: 619558)
[PASS] testAddVotingStrategiesInvalidAddress() (gas: 46379)
[PASS] testAddVotingStrategiesOverflow() (gas: 11245408)
[PASS] testCancel() (gas: 228707)
[PASS] testCancelAlreadyCancelled() (gas: 230288)
[PASS] testCancelAlreadyExecuted() (gas: 287997)
[PASS] testCancelInvalidProposal() (gas: 235014)
[PASS] testCancelUnauthorized() (gas: 227600)
[PASS] testRemoveAllVotingStrategies() (gas: 46168)
[PASS] testRenounceOwnership() (gas: 20260)
[PASS] testSetDaoURI() (gas: 57026)
[PASS] testSetMaxVotingDuration() (gas: 59824)
[PASS] testSetMaxVotingDurationInvalid() (gas: 63768)
[PASS] testSetMetadataURI() (gas: 48154)
[PASS] testSetMinVotingDelay() (gas: 59386)
[PASS] testSetMinVotingDurationInvalid() (gas: 50389)
[PASS] testSetProposalValidationStrategy() (gas: 56935)
[PASS] testSetVotingDelay() (gas: 73924)
[PASS] testSpaceUpgrade() (gas: 3358373)
[PASS] testSpaceUpgradeUnauthorized() (gas: 3326651)
[PASS] testTransferOwnership() (gas: 24921)
[PASS] testTransferOwnershipInvalid() (gas: 17298)
[PASS] testUpdateAuthenticators() (gas: 103421)
[PASS] testUpdateSettings() (gas: 93694)
[PASS] testUpdateSettingsUnauthorized() (gas: 44620)
[PASS] testUpdateStrategies() (gas: 184548)
[PASS] testUpdateStrategiesUnauthorized() (gas: 44181)
[PASS] testUpdateVotingStrategies() (gas: 116848)
[PASS] testUpdateVotingStrategiesArrayLengthMismatch() (gas: 37965)
Suite result: ok. 30 passed; 0 failed; 0 skipped; finished in 84.67ms (83.47ms CPU time)

Ran 4 tests for test/MerkleWhitelistVotingStrategy.t.sol:MerkleWhitelistVotingStrategyTest
[PASS] testLargeMerkleWhitelist() (gas: 7588260)
[PASS] testMerkleWhitelistInvalidMember() (gas: 29780)
[PASS] testMerkleWhitelistInvalidProof() (gas: 24229)
[PASS] testMerkleWhitelistVotingPower() (gas: 62096)
Suite result: ok. 4 passed; 0 failed; 0 skipped; finished in 136.63ms (123.58ms CPU time)

Ran 9 tests for test/OptimisticQuorum.t.sol:OptimisticTest
```

```
[PASS] testOptimisticQuorumEquality() (gas: 401838)
[PASS] testOptimisticQuorumLotsOfVotes() (gas: 17355866)
[PASS] testOptimisticQuorumMinVotingPeriodAccepted() (gas: 219601)
[PASS] testOptimisticQuorumMinVotingPeriodReached() (gas: 311341)
[PASS] testOptimisticQuorumNoVotes() (gas: 221153)
[PASS] testOptimisticQuorumOneVote() (gas: 292377)
[PASS] testOptimisticQuorumReached() (gas: 312182)
[PASS] testOptimisticQuorumSetQuorum() (gas: 338683)
[PASS] testOptimisticQuorumSetQuorumUnauthorized() (gas: 13691)
Suite result: ok. 9 passed; 0 failed; 0 skipped; finished in 71.00ms (65.52ms CPU time)

Ran 1 test for test/ForkedTests.t.sol:ForkedTest
[PASS] testFork_VoteAndProposeWithCompToken() (gas: 202495)
Suite result: ok. 1 passed; 0 failed; 0 skipped; finished in 3.72s (1.87s CPU time)

Ran 30 test suites in 3.73s (4.45s CPU time): 317 tests passed, 0 failed, 0 skipped (317 total tests)
```