2. FEBRUARY 2024

KAUZ SECURITY SERVICES

MOONWELL MULTICHAIN GOVERNANCE

TABLE OF CONTENTS

EXECUTIVE SUMMARY

This report presents the results of our engagement with Solidity Labs to review the Wormhole integration of the Moonwell v2 Multichain Governance smart contracts. Dominik Muhs conducted the review over one week, from **January 29, 2024, to February 02, 2024**. A total of 5 person-days were spent.

Overall, the main efforts focused on the Wormhole integration and the integrity of the voting process. On the in-scope code base, four low-severity findings were identified. One high-severity issue was removed after receiving clarifications from the Wormhole team.

SCOPE AND OBJECTIVES

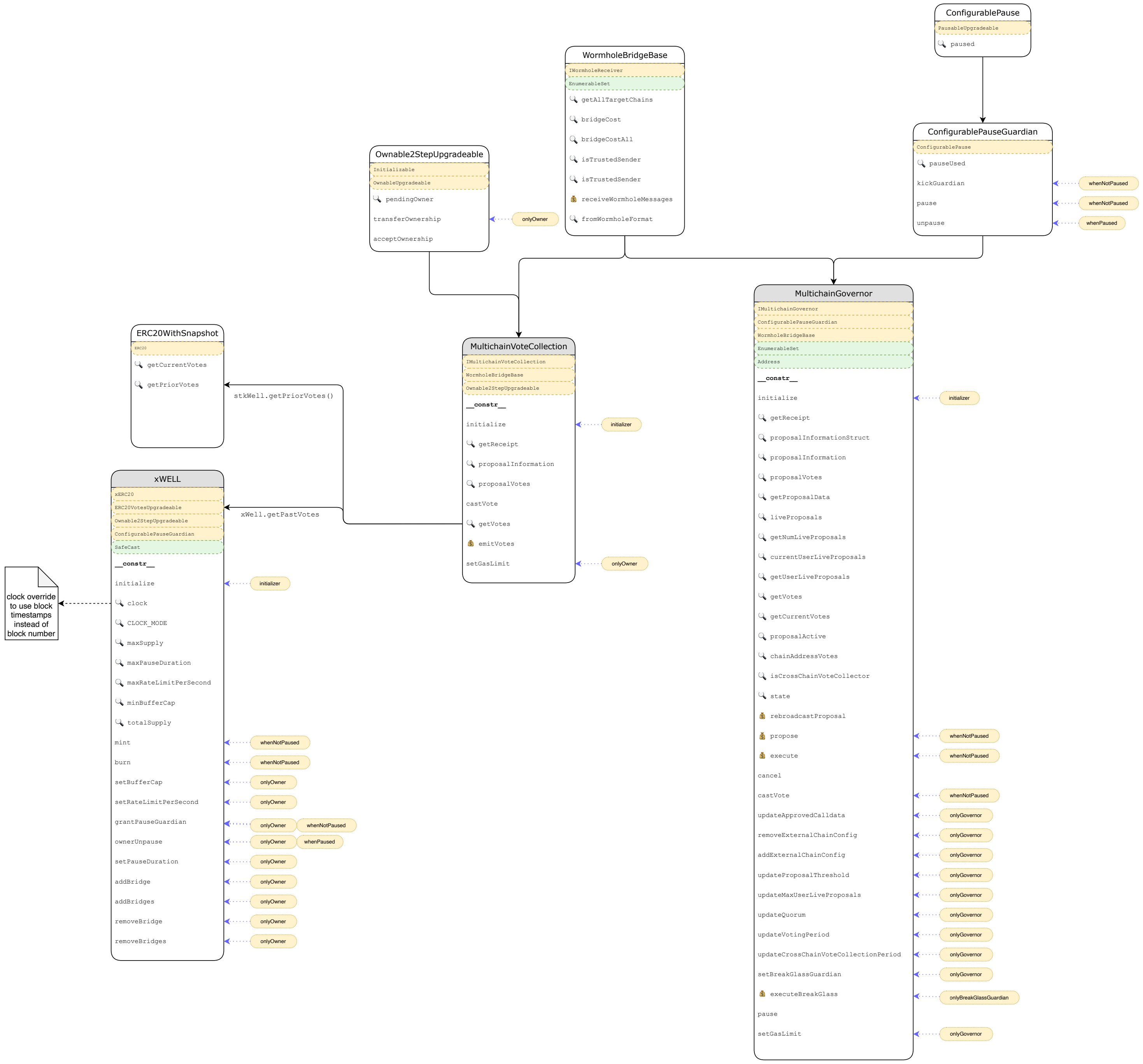
The review focused on the following commit hash: `e3aa4a555c9e8d4705b44568bd04eff897b0cdd4`, in pull request #101 on the target repository. On January 30, the scope was extended by the client to the following commit hash: `ebce011614b053674f75c147a4c188c59c333b13`. These additional changes have been partially covered.

Given the time constraints, together with the technical point of contact, it was agreed to conduct this review on a best-effort basis prioritizing the areas of concern:

- Resiliency against potentially malicious bridge messages,
- The consistency of stack components using block numbers and block timestamps,
- Potential inconsistencies stemming from multiple chains participating in the system, and
- Availability and reliability of the Wormhole Bridge infrastructure.

Furthermore, it's a priority to identify known vulnerabilities particular to smart contract systems, as outlined in the Smart Contract Security Field Guide and the Enterprise Ethereum Alliance's ETHTrust specification. A list of the files in scope can be found in Appendix 1.

AUDIT ARTIFACTS

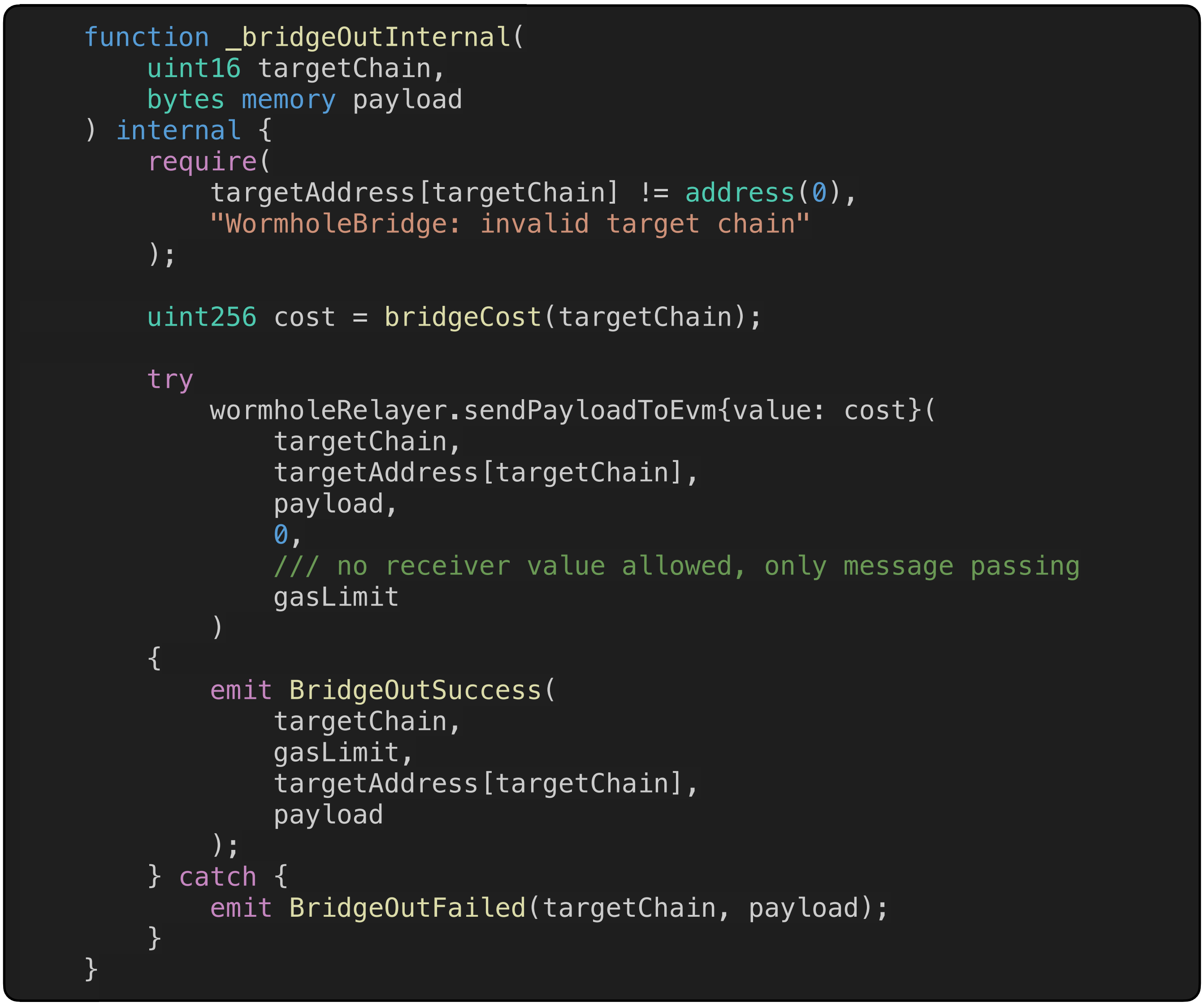


Architecture Diagram

FINDINGS

KSS-2: [LOW] REDUNDANT COST CALCULATION

Before sending messages to the bridge to be broadcast across the governance system, a cost estimate must be fetched and validated. This estimate is fetched in the `_bridgeOutInternal` function of the `WormholeBridgeBase` contract.



```
    function _bridgeOutInternal(
        uint16 targetChain,
        bytes memory payload
    ) internal {
        require(
            targetAddress[targetChain] != address(0),
            "WormholeBridge: invalid target chain"
        );

        uint256 cost = bridgeCost(targetChain);

        try
            wormholeRelayer.sendPayloadToEvm{value: cost}(
                targetChain,
                targetAddress[targetChain],
                payload,
                0,
                /// no receiver value allowed, only message passing
                gasLimit
            )
        {
            emit BridgeOutSuccess(
                targetChain,
                gasLimit,
                targetAddress[targetChain],
                payload
            );
        } catch {
            emit BridgeOutFailed(targetChain, payload);
        }
    }
```

Other functions leveraging this internal function are `_bridgeOut` and `_bridgeOutAll`. These functions also calculate their using `bridgeCost`, however.

RECOMMENDATION

The `_bridgeOutInternal` function can be optimized by adding a `cost` parameter to it, passing the previously calculated cost estimate.

KSS-3: [LOW] LAX PROPOSAL TIME VALIDATION

The `MultichainVoteCollection._bridgeIn` function aims to handle incoming Wormhole messages that have been previously validated through the `WormholeBridgeBase` contract. Once a relayer message is received, the `MultichainVoteCollection` contract creates a new proposal to start handling votes on the secondary chain.

The implementation in `_bridgeIn` relies implicitly on the validation performed by the `MultichainGovernor` contract's `propose` function. As such, it permits a broader range of values:

```
/// Ensure proposalId is unique
require(
    proposals[proposalId].votingStartTime == 0,
    "MultichainVoteCollection: proposal already exists"
);

/// Ensure votingSnapshotTime is less than votingStartTime
require(
    votingSnapshotTime < votingStartTime,
    "MultichainVoteCollection: snapshot time must be before start time"
);

/// Ensure votingStartTime is less than votingEndTime
require(
    votingStartTime < votingEndTime,
    "MultichainVoteCollection: start time must be before end time"
);

/// Ensure votingEndTime is in the future
require(
    votingEndTime > block.timestamp,
    "MultichainVoteCollection: end time must be in the future"
);
```

RECOMMENDATION

This implicit dependency between `MultichainVoteCollection` and `MultichainGovernor` proposal data should be avoided to make the code base more defensive and robust. Both smart contracts depend on the `Constants` library and can leverage its values as a shared source of truth. Concrete validation steps should mirror the same requirements as set during proposal creation. In general, trust in passed messages should be reduced.

KSS-4: [LOW] MISSING GAP VARIABLES

The majority of contracts in scope in the system are upgradeable. However, none of them contain gap variables to accommodate new storage slots introduced by future upgrades, making them more dangerous and error-prone.

RECOMMENDATION

A `__gap` array should be introduced and its size chosen such that it adds up to 50 with the number of storage slots the contract uses. See the OpenZeppelin and Solidity documentation pages on the topic for more information.

KSS-5: [LOW] UNUSED VARIABLE CAN RESULT IN AN INVALID ROLLBACK

The MultichainGovernor contract defines a `breakGlassGuardian`, which can execute authorized calldata once through the contract. It furthermore specifies a `governanceRollbackAddress` state variable holding an address to roll back to in case of emergency. It is assumed that this is done through authorized calldata that is part of the contract state.

```
    /// --------------------------------------------------------- ///
    /// ------------------------- SAFETY ------------------------ ///
    /// --------------------------------------------------------- ///

    /// @notice the governance rollback address
    address public override governanceRollbackAddress;

    /// @notice the break glass guardian address
    /// can only break glass one time, and then role is revoked
    /// and needs to be reinstated by governance
    address public override breakGlassGuardian;
```

However, this variable is unused and not initialized, thus containing 0x0. As there is no way to set it, any rollback payload relying on `governanceRollbackAddress` will either revert or brick the system, depending on the concrete mechanism used.

RECOMMENDATION

The state variable should either be removed if it is dead code or initialized properly if any out-of-scope functionality depends on it.

APPENDIX 1 - FILES IN SCOPE

All information relates to the following commit hash: e3aa4a555c9e8d4705b44568bd04eff897b0cdd4.

SHA-256	Name
52089369329c88d18bdd40f969e49b8b0b12b018a3af727bd520102ec9725f88	./4626/4626Deploy.sol
ad4a70bb91ffed731da1612ae81ba862c2f6e9b5cb79dfff686524cfb800c539	./4626/CompoundERC4626.sol
62e7df7fcb1ada35b5b463e9623b002a0a30413dedcf9230829b8db306658ce8	./4626/LibCompound.sol
6457f7c80fafaff49fc1b799b43bc9de6857f0a81f6f6c7b43ab68c5e134907e	./CarefulMath.sol
3a81f90514577a53eea4aebe39d9aa45bd46df3aae1f4cb00b0a6fc6fa9c2e21	./Comptroller.sol
556a8c76408647e54b6762a300ace6f56a53899a53a58fb0e23fce821a5c5030	./ComptrollerInterface.sol
18376e5bf19f28826e6ca824858ec298ccc24655e03eb750d682abfb62585173	./ComptrollerStorage.sol
ebf83bc323a460cd11726117340a6a5a8d76c7deadcb084dde295d2475030863	./EIP20Interface.sol
e6296c00805fd3665b9a9fdad855aa5922548bef480ff07278b302db3efc49d3	./EIP20NonStandardInterface.sol
f94e8ed0df445dced40155e0cb04312d9b7fc243c404b7849114f7927d1b39d5	./ErrorReporter.sol
d35828b410cf33195f458e0bb976649a4e16a56cf403d64d3d8cf52529ff7332	./Exponential.sol
2af74313be18fd8d616b2f2b2e1681b9852d5efd43c85b53760081b0278019bc	./ExponentialNoError.sol
67843e674fd1d55b9fbc81b2fd9dfd39b7637ddf0bd090c250b4e2777308b37d	./Governance/deprecated/MoonwellArtemisGovernor.sol
22d1a048f90c3728cfd0cc445ab551ee6098dbe3bf12d365db232e4911c474f1	./Governance/deprecated/Timelock.sol
157184b68285abbcfde8a2960911403a8af0e75b26c9b0fdee58a73dd6451367	./Governance/deprecated/Well.sol
90aad2331e46ecfcea55646f15b2bfe00ae9b189a5e7e81f1260779970ed36b4	./Governance/IERC20.sol
2816bc7f0e781007aab0c2b49642a040acd1ec57714ee0a200ed6e1b6455c648	./Governance/ITemporalGovernor.sol
9f323f319a886149734d7c0058f73fc8305c3ddabc34ce215d305cd772a75a94	./Governance/IWormholeTrustedSender.sol
b30ff2367f7f0a6235f6be06469afb34b123172af4b266381056126d83dbe7d5	./Governance/MultichainGovernor/Constants.sol
7dbbc0db8f243e82e3426ab5f4b0bb3154d04e6a54563169661af52fc89cbe72	./Governance/MultichainGovernor/IMultichainGovernor.sol
f69546dbdda804035938c36c7aa0aa89348508dca6a4b72f31b8ef4580098905	./Governance/MultichainGovernor/IMultichainVoteCollection.sol
13b70c1bf5c5a369d0156cb4b19aa5ed3269e35de234380e696105002377bf89	./Governance/MultichainGovernor/MultichainGovernor.sol

SHA-256	Name
e3bb0f5e9e5e614c3c0069c0e1e311f1f4da964a697d36d5820e683c707cc68d	./Governance/MultichainGovernor/MultichainGovernorDeploy.sol
80e7480a86d239d78ae03ec02cb5f0390deae9ce3c2842e9b50bbae9e570634d	./Governance/MultichainGovernor/MultichainVoteCollection.sol
05271eb6bc7166261a70504afc25ab2a3ae27e4cdda1ca1e85d211d4d6f58528	./Governance/MultichainGovernor/SnapshotInterface.sol
5c5609ed8303edd2fe774275fefce6e7a3cd6c16d36514380d00d632a7b61a3e	./Governance/TemporalGovernor.sol
157184b68285abbcfde8a2960911403a8af0e75b26c9b0fdee58a73dd6451367	./Governance/Well.sol
8f4642dd3cc07cf3554588dfa189eef69bc95832d661dbf75506a9efb4a60ce1	./Governance/WormholeTrustedSender.sol
b7c67161b63d976f43cb86b6eefe8f2d20da6c1f570b6c83ff1a480cb5ac5aa7	./IRModels/InterestRateModel.sol
0ea03f8cc38e635931a7b0dfa27e5074e9e1cea852622ac87067dc6b3378071e	./IRModels/JumpRateModel.sol
dea9cec65e83b14c43913eba3b834aceaa072854f263a9fcc06dd53cb04025b7	./IRModels/WhitePaperInterestRateModel.sol
768e17ec0e5c73a8dcf4db50e7256d1e2c7fc8c96550fd29cae9562f618701a4	./MErc20.sol
7805a81251c94ef7ef9f8aef38661ee271484dac3815ac9ab9fcc05c67746dc5	./MErc20Delegate.sol
1c7a315e91fa827d6a9856aafb9bdb184d933e30b4dd0b309cf80e31e615a821	./MErc20Delegator.sol
9649ce7711a423964bb48e2d85b23b8892ff059b9a0d1b3ffc19f3a63a19ff78	./MLikeDelegate.sol
2c9a2581e228af2dfc8464b703bf79ca17aa60d3cf365469930277d359e59042	./MToken.sol
229d790960d391d782019289305d9c0476e336a63e61f741f83de7b4c47e418a	./MTokenInterfaces.sol
8965db2075acae264364f533e3f417d58a1491814e9a55fd489b9e9c0f1038cc	./MultiRewardDistributor/MultiRewardDistributor.sol
519dbc48e6d1dd463daf4c0991bda9749d4f2036039aef224ddec8a62b4117f0	./MultiRewardDistributor/MultiRewardDistributorCommon.sol
3dbb0c73c5e7a1773310887e0a5a8dc3a99a5d84e52bf99d933e2d9901718235	./MWethDelegate.sol
2fa4237d16335a86651af254899be82d8ffdd93049f7b2d0188762460c1bef3d	./Oracles/AggregatorV3Interface.sol
9b177025a6502b45f084ef12dd69d131200a7d40ad13a5c4f4070f02f4b200ab	./Oracles/ChainlinkCompositeOracle.sol
69931d0e86a6b212903c43a8860d2823844fa47195c29857f350ec1231ed3300	./Oracles/ChainlinkOracle.sol
befd8d42a8c0fd5744f29afe967024211ddc87573e1008d491d6af575a16c972	./Oracles/PriceOracle.sol
9ec8d5fd329c8dea34e5814fa9ff4cdf180246a61ae8794e44cbc4858725d1bc	./proposals/Addresses.sol
fd0bc43787c595c176d0816384d8a6b7fdbb6f5bc7c6ca63a6f60052d5448d83	./proposals/Configs.sol
6b5ec9ea45af5637737f9ecf336123cce24ad6a80dc54528d1e062fa067b3693	./proposals/Deploy4626Vaults.s.sol
e04769110598698c34c04c3b550ac0f54bbfb6f3d0515a10cc47348c10f2a0d5	./proposals/DeployCompositeOracle.s.sol

SHA-256	Name
c64624aa56551cce735aa07a9d5bb6691dfa9ff985c49c8cc90e7129e7b95756	./proposals/DeployFaucetToken.s.sol
5fec7c47ca2866f9f75fe59cec0be5a5ffbe0276b885d2138def63b3811ccd08	./proposals/DeployJRM.s.sol
a47194c657d9457f6a88315aa9199ad266cdde224e002d19166c064147eac319	./proposals/DeployWETHRouter.s.sol
89a94ae69f7182f37c0036fdf9143e0b48527aa2140b00f98896896b27245ae2	./proposals/DeployWethUnwrapper.s.sol
596eb9bd49708e754fe8c7f8ad76c746001224913a3767a3692614d1a6fc4d83	./proposals/hooks/MarketCreationHook.sol
8ba9d28b19d4094bc37ce9296bba532702f6064b31d86bdb88b3291e08f69196	./proposals/MIPProposal.s.sol
c4f5dd78d1559c73a656ee37e864690bcefc4f850cabbcb6ee4953a6b6b22a9d	./proposals/mips/examples/mip-market-listing/mip-market-listing.sol
384bf8b39d3a243293bd2395b9b3dcb2235ad51ab86f53d59be712bb105b4217	./proposals/mips/mip-b00/mip-b00.sol
3212ef0dcb845137ae60089866c66d8bc3b84cafb174da12f6fe1e47d30993b2	./proposals/mips/mip-b01/mip-b01.sol
52676d7c042a9dd272de642bb4e4e953d71bfb07188c90689ca6adc520d5e202	./proposals/mips/mip-b02/mip-b02.sol
a8c6d4e6dcddb8f0a8a737e4f4a83b8509c81c97c198a3cc7113561773e5a61b	./proposals/mips/mip-b03/mip-b03.sol
f253d0ecad7ed1861f7d10bfa7740051bcfac6ee15e195eb089a8a6f468fa8c6	./proposals/mips/mip-b05/mip-b05.sol
7861394ed45ddae208b95da72990e425d5d1d56b6ab1dd5ddf1f6b2e42bd16bd	./proposals/mips/mip-b06/mip-b06.sol
3a38ca55ab3c8687d9b2a590a1a88765a5c24ba7cb2e0bbc96dfb8768a7243ec	./proposals/mips/mip-b07/mip-b07.sol
07147957a40ad86f1756a2e4095136c8d49197fe2b335cfae3889e59776f79ee	./proposals/mips/mip-b08/mip-b08.sol
9c5ee8bfccfd429372935fd76a8811a2bf1a40ce523cb8238d4127e583efd59c	./proposals/mips/mip-b09/mip-b09.sol
e84da72be7715d44164249cd6123f53df88beddf0d3419be105626daf08c11a1	./proposals/mips/mip-b10/mip-b10.sol
9a717493781a32cc1b31854fa657908a9015b70ac642fe0bb0b61dd84df59ae7	./proposals/mips/mip-b11/mip-b11.sol
67ea477101bc8c842f03d05e12f682c3e045a6e2bf51fb70e9d56d092d30314e	./proposals/mips/mip-b12/mip-b12.sol
0b26ab0b953513c37d101fe84202784b26973ed74f74abe9825b9dd599848deb	./proposals/mips/mip-m16/mip-m16.sol
dd9a6f328cf610b1c03b3682ab14bed941ecd9049e0110f41e0225bd94249b8b	./proposals/mips/mip-t01/mip-t01.sol
2bc91f490e5c95f7ab23c783a180f4da53a9f88a55c8d0852be0b8f5416774a2	./proposals/mips/mip-xwell/xwellDeployBase.sol
f001922d8b6214ba6853df0f434c55f2bf28897281d083c9695ab04217e9ae96	./proposals/mips/mip-xwell/xwellDeployMoonbeam.sol
d9bba4c4533d66a372c34024ecb0d7fa59ead2250794d324e985be84d0c8c88b	./proposals/proposalTypes/CrossChainJSONProposal.sol
09f42113cac57fe9abe3c56df88d1311c567abea3c14d67ac12503ffebcc98fa	./proposals/proposalTypes/CrossChainProposal.sol
190fe577fd64bbc8a78d1c88e9dd37b70976e9fc9c427e89af0d144d2d9d41e3	./proposals/proposalTypes/GovernanceProposal.sol

SHA-256	Name
699631c485196f5df98a08eb900b1532e76b1b76ddb9d14ee5d5f31e0dcabc6a	./proposals/proposalTypes/MultisigProposal.sol
350df559701dc9c4eb42ead5390bb0cd0277634c31ec974eb7ea9a27934607ec	./proposals/proposalTypes/Proposal.sol
353e3e7f57cb3b1cbb89dd0cc23b7eba487e3ecae6151960c001d657c0cd43ff	./proposals/proposalTypes/TimelockProposal.sol
f64f435fc485233b750ba5fc707cae36d187b9224e442656c96a52ada1b39883	./proposals/TestProposals.sol
d229fd5596d26d45fc305b9083fcd9c7414c79a17648a9c5dace3e91820b8b7f	./proposals/utils/CreateCode.sol
0d9abf01f447aae70a5865e167796ea1e0be0d814f1836c31a0aab049a3b8437	./proposals/utils/ParameterValidation.sol
e465d260203ff29b64f7a80e84ae45cbe99f3a7bfa8d983c147f843005ecb8e8	./proposals/utils/StringUtils.sol
c0d140109272b28f1cf180138f349dd8c8c0cf3d389aa710dd48195c5278a2ad	./Recovery.sol
3093ae61e6f7e67e5bd192ec478bb3992988942c146b09a3fe28118e32ccd3c0	./router/IWETH.sol
2f7455c00f7c9b3dec937e8fe9a1d0f8f8c1ebb34b0afc5939ece58923625c84	./router/WETHRouter.sol
4649554a3909a6854382a98239cb07cdc4161839418ef33d61325ab12e7ac855	./SafeMath.sol
0bb56fdd6af5ab71e4279405aa6e3831b8c794832dfa2f72feaa6b3278d9a39e	./stkWell/src/Address.sol
7e61482a46a078f787ef9e81254340d0b06656fd3a4ea8f6261d5f84c171578d	./stkWell/src/Context.sol
49686fdbd6f0bb1b378294de793096752a183060533a1cfe63fdca4442535ea3	./stkWell/src/DistributionManager.sol
6a249dda93000b22c207d6cc7c2e0b34fb2c9b2dc5227f33ffe82e50c011ccb1	./stkWell/src/DistributionTypes.sol
cd4c0bbf6356b38b320f12674c15f104f9eba54b885a674e23cfb41b56b7ea1f	./stkWell/src/ERC20.sol
8b0fb744ee192970cbedd1bd72d74e63ab758a492431cf8d91ad05f7738cc115	./stkWell/src/ERC20WithSnapshot.sol
846672b06591f917ee7e7a7e16f7e915befa99ae7d72b89cd776c861f2761cfe	./stkWell/src/Initializable.sol
3af3e638e635379b65e8e0b7d1cf6cdf6c4364df60c567b31196f5abdf093add	./stkWell/src/ReentrancyGuardUpgradeable.sol
4ec7b48adc5cf84faaddae8d5f4e15fe91c92f97dd670ca60bba3bee0209db60	./stkWell/src/SafeERC20.sol
14f57fce235cc8219c799ccedc5626ad356aa98e52e35419f18d1ac9394612b9	./stkWell/src/SafeMath.sol
dfdd015bd1634fbc0636808381855a1df4e837334358a52b94869bd5a59c50f6	./stkWell/src/StakedToken.sol
c1372b391efb599d0a317aadf32e799b93e8bcda08e2813c35806e231c70f43f	./stkWell/src/StakedWell.sol
1d4e99f1cc333b308089c997f5a6ac04298dbfe25221b22e4640aba1289524c2	./Unitroller.sol
2a98780f18d827ab1af1267a4bbef20b2f48691b0f49317f0396b3a8de496be5	./views/BaseMoonwellViews.sol
fc48a1c81764b34213ea077dc7ad4b39e09022fcb2413f8a52634b54b79c434c	./views/ComptrollerInterfaceV1.sol
46e96cc2ea8cd7fef0954e2ecbb36d304d3d221f5ebd36903933d399343ac388	./views/MoonwellViewsV1.sol
e7eff34935f9ce67905000c81bd9e5d7e92248386cbf1354a93da8a2bea46db6	./views/MoonwellViewsV2.sol

SHA-256	Name
d42d1c25be7ba856585b5cd61d45fe50f5dc4bb9160f8503165a2758ccb91144	./views/SafetyModuleInterfaceV1.sol
54eddfee07d399a500f0288ea3665b77617c5d8f61e04520ea6c17aa5554218b	./views/TokenSaleDistributorInterfaceV1.sol
a2e72c0dab358d5594c3f7e2c73541109d18d6f79dbeadbfc66a66774c506f89	./views/UniswapV2PairInterface.sol
b86cc3a50f7aeba7bdf98830d91fb8da89fa58578d339e0d5126137b310e018b	./WethUnwrapper.sol
ce955f34f981f48c55e8c2e4230989dc8aa221c15788a455f41b7ae3023eb1f6	./wormhole/WormholeBridgeBase.sol
c5491e837aa7d30ea07e65a0dae4439f39fb54154a8f99a10dec5b82279fa2b4	./xWELL/AxelarBridgeAdapter.sol
fe5476f35f95f64c72aaf4f004b73376bb08de76c3ce7298ca11f101da60fa0b	./xWELL/ConfigurablePause.sol
7b8681effe541129f9361bd1b40cd52d72c047b64ddf45c1ffdc575dfb6c2452	./xWELL/ConfigurablePauseGuardian.sol
15bbc2b5e69a73d16111a2853431aff22287071e414d8b07991f438865072dd6	./xWELL/MintLimits.sol
caf783d134fe98cd4b2dd71a7b8e4d59f2e83892e8585cf03a16f422bae76c52	./xWELL/WormholeBridgeAdapter.sol
10bb9ab115e0bcaed386e676e6e218f090f3c8462b561a89bb0a252531618502	./xWELL/xERC20.sol
e18ae14bf4413370c5c03cf9b2f6757eb8341ead614168698f2144c3b4c80651	./xWELL/xERC20BridgeAdapter.sol
9ee4e20ee820c3069b09f881966ef80f16adf915c72efec462c94bb92d2774bf	./xWELL/XERC20Lockbox.sol
6c133139af5d58073885e81ee510bfc498dc05123f46bffa4bf72a9ac950ce2d	./xWELL/xWELL.sol
004ed6a65f29e2de90418ab1930a66d83dfc456878ab19c85a94663a16fd518b	./xWELL/xWELLDeploy.sol
976af8b8337582a41edac59835b10aa84699ce70cf91d144fa4f5a45e87f7d53	./xWELL/xWELLRouter.sol

APPENDIX 2 - DISCLAIMER

Kauz Security Services ("KSS") typically receives compensation from one or more clients (the "Clients") for performing the analysis contained in these reports (the "Reports"). The Reports may be distributed through other means, including via KSS publications and other distributions.

The Reports are not an endorsement or indictment of any particular project or team, and the Reports 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 with high levels of technical risk and uncertainty. No Report provides any warranty or representation to any Third Party in any respect, including regarding the bug-free 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 Reports in any way, including for the purpose of making any decisions to buy or sell any token, product, service, or other asset. Specifically, to avoid doubt, 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 is not a guarantee as to the absolute security of the project. KSS owes no duty to any Third Party by publishing these Reports.

The Reports 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 code and only the code we note as being within the scope of our review within this report. Any Solidity code itself presents unique and unquantifiable risks as the Solidity language itself remains under development and is subject to unknown risks and flaws. The review does not extend to the compiler layer, development tools, or any other areas beyond specified code that could present security risks. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. In some instances, we may perform penetration testing or infrastructure assessments depending on the scope of the particular engagement.

You may, through hypertext or other computer links, gain access to websites operated by persons other than KSS. Such hyperlinks are provided for your reference and convenience only and are the exclusive responsibility of such websites' owners. You agree that KSS is not responsible for the content or operation of such websites and that KSS shall have no liability to you or any other person or entity for the use of third-party websites. Except as described below, a hyperlink from this website to another website does not imply or mean that KSS endorses the content on that website or the operator or operations of that site. You are solely responsible for determining the extent to which you may use any content at any other websites you link from the Reports. KSS assumes no responsibility for the use of third-party software on the website and shall have no liability whatsoever to any person or entity for the accuracy or completeness of any outcome generated by such software.

The content contained in the Reports is current as of the date appearing on the Report and is subject to change without notice. Unless indicated otherwise, by KSS.