OmniBridge

SMART CONTRACT AUDIT

06.05.2021

Made in Germany by Chainsulting.de

Table of contents

1. Disclaimer

The audit makes no statements or warrantees about utility of the code, safety of the code, suitability of the business model, investment advice, endorsement of the platform or its products, 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 information presented in this report is confidential and privileged. If you are reading this report, you agree to keep it confidential, not to copy, disclose or disseminate without the agreement of POA Network. If you are not the intended receptor of this document, remember that any disclosure, copying or dissemination of it is forbidden.

Major Versions / Date	Description
0.1 (20.03.2021)	Layout
0.5 (30.03.2021)	Automated Security Testing Manual Security Testing
0.6 (14.04.2021)	Verify Claims and Test Deployment
0.8 (15.04.2021)	Unit Tests
0.9 (16.04.2021)	Summary and Recommendation
1.0 (16.04.2021)	Final document
1.1 (05.05.2021)	re-checking contracts
1.2 (06.05.2021)	Adding github commits

2. About the Project and Company

Company address:

POA Networks Ltd
Genesis Building 5th Floor
Genesis Close PO Box 446
George Town, Grand Cayman KY1-1106
Cayman Islands

Website: https://www.poa.network

Twitter: https://twitter.com/poanetwork

Reddit: https://www.reddit.com/r/POA/new

Telegram: https://t.me/poa_network

Youtube: https://www.youtube.com/channel/UCPp7CZ1OPvBVhHOb8CDxU2A

GitHub: https://github.com/poanetwork

Discord: https://discord.gg/mPJ9zkq

Blog: https://medium.com/@poanetwork

2.1 Project Overview

POA Core is an autonomous network secured by a group of trusted validators. All validators on the network are United States notaries, and their information is publicly available. This distributed group of known validators allows the network to provide fast and inexpensive transactions. POA organization also develops products and tools to improve interoperability, infrastructure and transparency throughout the ecosystem. These include BlockScout, an open-source explorer, TokenBridge, a multi-chain asset-transfer solution.

The OmniBridge multi-token extension for the Arbitrary Message Bridge between Ethereum and the xDai chain is the simplest way to transfer ANY ERC20/ERC677/ERC827 token to the xDai chain. By using OmniBridge any user (not only the token contract owner) can transfer tokens from Ethereum to a chain with fast, inexpensive transactions (in this case the xDai chain) without deploying any additional contracts. The specified token amount is locked in the mediator contract, a new token contract is deployed automatically on the xDai chain, and the requested token amount is minted on the xDai chain. The reverse operation burns bridgeable tokens on the xDai chain and unlocks the tokens from the token contract on Ethereum.

3. Vulnerability & Risk Level

Risk represents the probability that a certain source-threat will exploit vulnerability, and the impact of that event on the organization or system. Risk Level is computed based on CVSS version 3.0.

Level	Value	Vulnerability	Risk (Required Action)
Critical	9 – 10	A vulnerability that can disrupt the contract functioning in a number of scenarios, or creates a risk that the contract may be broken.	Immediate action to reduce risk level.
High	7 – 8.9	A vulnerability that affects the desired outcome when using a contract, or provides the opportunity to use a contract in an unintended way.	Implementation of corrective actions as soon as possible.
Medium	4 – 6.9	A vulnerability that could affect the desired outcome of executing the contract in a specific scenario.	Implementation of corrective actions in a certain period.
Low	2 – 3.9	A vulnerability that does not have a significant impact on possible scenarios for the use of the contract and is probably subjective.	Implementation of certain corrective actions or accepting the risk.
Informational	0 – 1.9	A vulnerability that have informational character but is not effecting any of the code.	An observation that does not determine a level of risk

4. Auditing Strategy and Techniques Applied

Throughout the review process, care was taken to evaluate the repository for security-related issues, code quality, and adherence to specification and best practices. To do so, reviewed line-by-line by our team of expert pentesters and smart contract developers, documenting any issues as there were discovered.

4.1 Methodology

The auditing process follows a routine series of steps:

1. Code review that includes the following:
   i. Review of the specifications, sources, and instructions provided to Chainsulting to make sure we understand the size, scope, and functionality of the smart contract.
   ii. Manual review of code, which is the process of reading source code line-by-line in an attempt to identify potential vulnerabilities.
   iii. Comparison to specification, which is the process of checking whether the code does what the specifications, sources, and instructions provided to Chainsulting describe.
2. Testing and automated analysis that includes the following:
   i. Test coverage analysis, which is the process of determining whether the test cases are actually covering the code and how much code is exercised when we run those test cases.
   ii. Symbolic execution, which is analysing a program to determine what inputs causes each part of a program to execute.
3. Best practices review, which is a review of the smart contracts to improve efficiency, effectiveness, clarify, maintainability, security, and control based on the established industry and academic practices, recommendations, and research.
4. Specific, itemized, actionable recommendations to help you take steps to secure your smart contracts.

4.2 Used Code from other Frameworks/Smart Contracts (direct imports)

Dependency / Import Path	Source
@openzeppelin/contracts/token/ERC20/IERC20.sol	https://github.com/OpenZeppelin/openzeppelin-contracts/blob/v3.2.2-solc-0.7/contracts/token/ERC20/IERC20.sol
@openzeppelin/contracts/utils/Address.sol	https://github.com/OpenZeppelin/openzeppelin-contracts/blob/v3.2.2-solc-0.7/contracts/utils/Address.sol
@openzeppelin/contracts/math/SafeMath.sol	https://github.com/OpenZeppelin/openzeppelin-contracts/blob/v3.2.2-solc-0.7/contracts/math/SafeMath.sol
@openzeppelin/contracts/token/ERC20/SafeERC20.sol	https://github.com/OpenZeppelin/openzeppelin-contracts/blob/v3.2.2-solc-0.7/contracts/token/ERC20/SafeERC20.sol
@openzeppelin/contracts/token/ERC20/ERC20.sol	https://github.com/OpenZeppelin/openzeppelin-contracts/blob/v3.2.2-solc-0.7/contracts/token/ERC20/ERC20.sol

4.3 Tested Contract Files

The following are the MD5 hashes of the reviewed files. A file with a different MD5 hash has been modified, intentionally or otherwise, after the security review. You are cautioned that a different MD5 hash could be (but is not necessarily) an indication of a changed condition or potential vulnerability that was not within the scope of the review

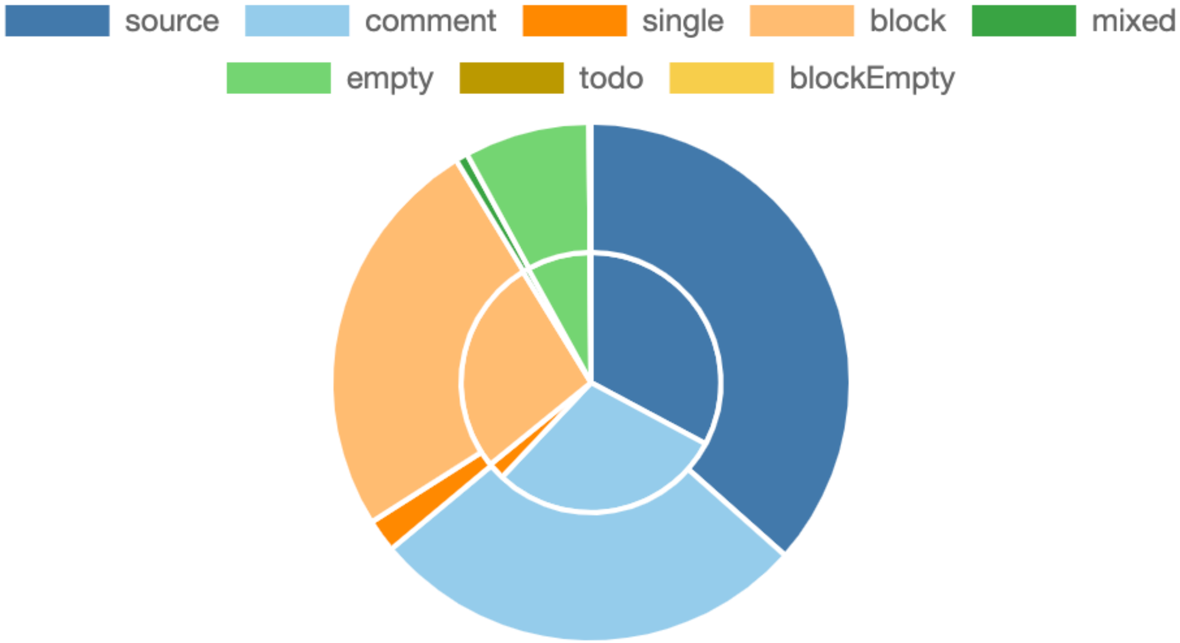
File	Fingerprint (MD5)
upgradeability/UpgradeabilityStorage.sol	ecc599c4896b113f2f86a654901de0b4
upgradeability/UpgradeabilityProxy.sol	e38e90d894797fec00158a08b7f08972
upgradeability/EternalStorage.sol	d5dfe5ec64e03d46551c4a478d517eb3
upgradeability/UpgradeabilityOwnerStorage.sol	da95982ecfb4f557d4e09730ab89ab9c
upgradeability/Proxy.sol	71ca7a9bd61cabc0da389218ab679075
upgradeability/EternalStorageProxy.sol	fbd47471d19d1984c4f4daec413c0bb8
upgradeability/OwnedUpgradeabilityProxy.sol	17045a54170dfe169d7bb73e3805d74e
interfaces/IERC677.sol	984900f25b304e027e9501dabb4859e7
interfaces/IOmnibridge.sol	d58bc7cfba4beb8470edeb13ed8408ee
interfaces/IUpgradeabilityOwnerStorage.sol	f18124c47fb8bc5d8fe102bc880d9e51
interfaces/IERC20Receiver.sol	675a97b20be142b7c934c1011496bd73
interfaces/IERC20Metadata.sol	c5e1d2a9eca735e59ab9b3c7fb7138fd
interfaces/IBurnableMintableERC677Token.sol	fdacb3c22dfd2b1c11cb7226a1dc596a
interfaces/IWETH.sol	a7d82ff8c8a21c9a279698883c5f93cd
interfaces/IAMB.sol	407c47468d876315b96dca0f2e42f7a7
helpers/WETHOmnibridgeRouter.sol	b6be6d0ae91e464af3fabffe09963c5b
libraries/TokenReader.sol	448474df8756c49bf19aa94255fded75
libraries/Bytes.sol	0d0760014133ebb0edfc2fe494a9425f
libraries/AddressHelper.sol	3d0f85a8a77aa4d80845cb022b7128b0
upgradeable_contracts/ForeignOmnibridge.sol	e5f563afa4781b93d7a3f60aa70ed47d
upgradeable_contracts/BasicOmnibridge.sol	c890dc82ab8d2c66021594f78d5b0652
upgradeable_contracts/modules/MediatorOwnableModule.sol	986d7ff6ead461a74f5f5da229822fc6
upgradeable_contracts/HomeOmnibridge.sol	d64090f1e2782fcb8add560de9ccc935
upgradeable_contracts/Ownable.sol	62c3ea6c4e0780238c7aecfaf9a7bf20

upgradeable_contracts/Sacrifice.sol	47a511372534ea2414db668ae61707d1
upgradeable_contracts/VersionableBridge.sol	48741ea3b05d6dee01679d4bbafcf89a
upgradeable_contracts/BasicAMBMediator.sol	5823e4ff3cd1dec313e98ac68ab9a3d1
upgradeable_contracts/Upgradeable.sol	ef561249275d05ec37a9ded90c9f1e4c
upgradeable_contracts/ReentrancyGuard.sol	ccd3ef2233cf55f5b0412d4142cfeeef
upgradeable_contracts/Claimable.sol	99583dab4834126ea4f5531e43b3ce68
upgradeable_contracts/Initializable.sol	169d9ef64fcd192d9d7bf48c3253a446
upgradeable_contracts/modules/OwnableModule.sol	5f221007bdb76c850f5347ddd14519a2
upgradeable_contracts/modules/gaslimit/SelectorTokenGasLimitManager.sol	32afc5ba7f2f73b99f8c1c4505bf452e
upgradeable_contracts/modules/gaslimit/SelectorTokenGasLimitConnector.sol	35b0aeadfabcd7f36f60184dbbd251ba
upgradeable_contracts/components/bridged/BridgedTokensRegistry.sol	1b643e511688474e59f7ad8ca69fa799
upgradeable_contracts/modules/factory/TokenProxy.sol	441bc1437c72f514fc06ef1de27e52b9
upgradeable_contracts/modules/feemanager/OmnibridgeFeeManager.sol	138a89dcb7d4665393734900b5753495
upgradeable_contracts/modules/forwarding_rules/MultiTokenForwardingRulesConnector.sol	0463be93255bbdb0be4fd0bdc522198b
upgradeable_contracts/modules/factory/TokenFactory.sol	68a7a5043e6463139fb463f992616779
upgradeable_contracts/modules/fee_manager/OmnibridgeFeeManagerConnector.sol	f7e20e6caa597733efa901ece455563d
upgradeable_contracts/modules/factory/TokenFactoryConnector.sol	c764ea34f0d05332d22a512d3e3b89ff
upgradeable_contracts/modules/forwarding_rules/MultiTokenForwardingRulesManager.sol	7e0b6f4eaea258c17083d1665976ad6d
upgradeable_contracts/components/common/FailedMessagesProcessor.sol	be58d62195d0bdbf11a54fad6569b9a4
upgradeable_contracts/components/common/GasLimitManager.sol	05fcabd264f547aceec36eebfd55be7b
upgradeable_contracts/components/common/BridgeOperationsStorage.sol	d6ad039a39f34c073777916761919826
upgradeable_contracts/components/common/TokensBridgeLimits.sol	686910a0b5e2704cbda8f5010a447912
upgradeable_contracts/components/common/TokensRelayer.sol	69440c3d555aa69c9b75424ee22057b9
upgradeable_contracts/components/common/OmnibridgeInfo.sol	522a70271f7aa19b24b453bba6118379
upgradeable_contracts/components/native/MediatorBalanceStorage.sol	7711f25aebfc34b9a61c50f4711e63bc
upgradeable_contracts/components/native/NativeTokensRegistry.sol	5124b0ea16a9e1e7526ee0508ba653eb

4.4 Metrics / CallGraph

Full Version: https://chainsulting.de/wp-content/uploads/2021/04/poa_solidity-metrics.html

4.5 Metrics / Source Lines



4.6 Metrics / Capabilities

Solidity Versions observed	🧪 Experimental Features	💰 Can Receive Funds	🖥 Uses Assembly	💣 Has Destroyable Contracts
`0.7.5`		`yes`	`yes` (11 asm blocks)	`yes`

📤 Transfers ETH	⚡ Low-Level Calls	👥 DelegateCall	🧮 Uses Hash Functions	🔖 ECRecover	🌀 New/Create/Create2
`yes`		`yes`	`yes`		`yes` → `NewContract:TokenProxy`

🌐Public	💰Payable
144	7

External	Internal	Private	Pure	View
100	198	1	11	85

StateVariables

Total	🌐Public
50	9

4.7 Metrics / Source Unites in Scope

Type	File	Logic Contracts	Interfaces	Lines	nLines	nSLOC	Comment Lines	Complex. Score	Capabilities
📝	contracts/upgradeability/UpgradeabilityStorage.sol	1		29	29	11	14	7	
📝	contracts/upgradeability/UpgradeabilityProxy.sol	1		46	46	18	21	13	
📝	contracts/upgradeability/EternalStorage.sol	1		14	14	9	4	7	
📝	contracts/upgradeability/UpgradeabilityOwnerStorage.sol	1		25	25	10	12	4	
🎨	contracts/upgradeability/Proxy.sol	1		95	89	21	70	63	🖥💰👥
📝	contracts/upgradeability/EternalStorageProxy.sol	1		15	15	5	7	5	
📝	contracts/upgradeability/OwnedUpgradeabilityProxy.sol	1		70	66	26	33	30	💰
🔍	contracts/interfaces/IERC677.sol		1	17	8	5		9	
🔍	contracts/interfaces/IOmnibridge.sol		1	9	4	3		3	
🔍	contracts/interfaces/IUpgradeabilityOwnerStorage.sol		1	5	4	3		3	🔆
🔍	contracts/interfaces/IERC20Receiver.sol		1	9	4	3		3	
🔍	contracts/interfaces/IERC20Metadata.sol		1	9	4	3		7	

Type	File	Logic Contracts	Interfaces	Lines	nLines	nSLOC	Comment Lines	Complex. Score	Capabilities
🔍	contracts/interfaces/IBurnableMintableERC677Token.sol		1	11	6	4		9	
🔍	contracts/interfaces/IWETH.sol		1	9	4	3		10	💰
🔍	contracts/interfaces/IAMB.sol		1	47	14	12		27	
📝	contracts/helpers/WETHOmnibridgeRouter.sol	1		98	94	43	41	49	🖥💰
📚🔍	contracts/libraries/TokenReader.sol	1	1	99	86	53	37	80	🖥
📚	contracts/libraries/Bytes.sol	1		22	22	8	13	9	🖥
📚	contracts/libraries/AddressHelper.sol	1		20	20	9	9	14	📤
📝	contracts/upgradeable_contracts/ForeignOmnibridge.sol	1		165	140	62	60	68	
🎨	contracts/upgradeable_contracts/BasicOmnibridge.sol	1		463	370	192	142	180	
📝	contracts/upgradeable_contracts/modules/MediatorOwnableModule.sol	1		30	30	14	12	8	
📝	contracts/upgradeable_contracts/HomeOmnibridge.sol	1		222	191	91	76	91	🧮
📝	contracts/upgradeable_contracts/Ownable.sol	1		74	74	35	34	24	

Type	File	Logic Contracts	Interfaces	Lines	nLines	nSLOC	Comment Lines	Complex. Score	Capabilities
📝	contracts/upgradeable_contracts/Sacrifice.sol	1		7	7	6		5	💰💣
🔍	contracts/upgradeable_contracts/VersionableBridge.sol		1	14	4	3		5	
🎨	contracts/upgradeable_contracts/BasicAMBMediator.sol	1		100	97	42	44	31	
📝	contracts/upgradeable_contracts/Upgradeable.sol	1		11	11	8	1	5	
📝	contracts/upgradeable_contracts/ReentrancyGuard.sol	1		21	21	13	8	11	🖥
📝	contracts/upgradeable_contracts/Claimable.sol	1		54	54	26	22	15	
📝	contracts/upgradeable_contracts/Initializable.sol	1		15	15	11	1	6	
📝	contracts/upgradeable_contracts/modules/OwnableModule.sol	1		35	35	15	15	6	
📝	contracts/upgradeable*contracts/modules/gas*limit/SelectorTokenGasLimitManager.sol	1		206	194	94	84	80	🖥
🎨	contracts/upgradeable*contracts/modules/gas*limit/SelectorTokenGasLimitConnector.sol	1		48	48	26	17	23	
📝	contracts/upgradeable_contracts/components/bridged/BridgedTokensRegistry.sol	1		41	41	16	19	16	🧮

Type	File	Logic Contracts	Interfaces	Lines	nLines	nSLOC	Comment Lines	Complex. Score	Capabilities
📝🔍	contracts/upgradeable_contracts/modules/factory/TokenProxy.sol	1	1	80	73	49	24	40	🖥🧮
📝	contracts/upgradeable*contracts/modules/fee*manager/OmnibridgeFeeManager.sol	1		242	230	114	95	102	📤
📝	contracts/upgradeable*contracts/modules/forwarding*rules/MultiTokenForwardingRulesConnector.sol	1		55	51	22	24	18	
📝	contracts/upgradeable_contracts/modules/factory/TokenFactory.sol	1		48	43	16	22	24	🌀
🎨	contracts/upgradeable*contracts/modules/fee*manager/OmnibridgeFeeManagerConnector.sol	1		88	81	43	32	36	📤
📝	contracts/upgradeable_contracts/modules/factory/TokenFactoryConnector.sol	1		39	39	18	17	13	
📝	contracts/upgradeable*contracts/modules/forwarding*rules/MultiTokenForwardingRulesManager.sol	1		143	126	48	70	51	
🎨	contracts/upgradeable_contracts/components/common/FailedMessagesProcessor.sol	1		65	60	29	22	37	🧮
🎨	contracts/upgradeable_contracts/components/common/GasLimitManager.sol	1		48	48	19	24	13	
🎨	contracts/upgradeable_contracts/components/common/BridgeOperationsStorage.sol	1		60	60	22	31	15	🧮

Type	File	Logic Contracts	Interfaces	Lines	nLines	nSLOC	Comment Lines	Complex. Score	Capabilities
📝	contracts/upgradeable_contracts/components/common/TokensBridgeLimits.sol	1		318	310	148	135	174	🧮
🎨	contracts/upgradeable_contracts/components/common/TokensRelayer.sol	1		126	101	49	43	102	🖥
📝	contracts/upgradeable_contracts/components/common/OmnibridgeInfo.sol	1		44	35	17	15	7	
📝	contracts/upgradeable_contracts/components/native/MediatorBalanceStorage.sol	1		28	28	11	14	9	🧮
📝	contracts/upgradeable_contracts/components/native/NativeTokensRegistry.sol	1		28	28	12	13	12	🧮
📝📚🔍🎨	**Totals**	**41**	**11**	**3567**	**3199**	**1520**	**1377**	**1579**	🖥💰💣📤👥🧮🌀🔆

Legend: [➖]

- **Lines**: total lines of the source unit
- **nLines**: normalized lines of the source unit (e.g. normalizes functions spanning multiple lines)
- **nSLOC**: normalized source lines of code (only source-code lines; no comments, no blank lines)
- **Comment Lines**: lines containing single or block comments
- **Complexity Score**: a custom complexity score derived from code statements that are known to introduce code complexity (branches, loops, calls, external interfaces, ...

5. Scope of Work

The POA Network Team provided us with the files that needs to be tested. The scope of the audit are the OmniBridge contracts.

Following contracts with the direct imports has been tested:

- BasicOmnibridge.sol
- ForeignOmnibridge.sol
- HomeOmnibridge.sol

The team put forward the following assumptions regarding the security, usage of the contracts:

1. **Background:** A user is able to send ERC20/ERC677- compatible tokens to the OmniBridge (OB) by calling either the relayTokens/relayTokensAndCall method of the OB contract (the OB contract must be approved in advance by the user) or the transferAndCall method of the ERC677 token. Depending on the nature of the tokens they can be either locked on the OmniBridge contract if the token natively deployed on a chain where one of the bridge contracts is deployed to, or burnt if the token has been deployed by the OB Token Fabric as part of the bridging process initiated beforehand. Some fraction of tokens can be distributed among the bridge fee recipient instead of locking/burning. Eventually, the OB contract pass a message to the Arbitrary Message Bridge (AMB) contract to deliver a bridge tokens request to another side of the bridge.
   Claim: There is no such case when tokens were not actually locked/burnt but the message for AMB to deliver a bridge tokens request was sent.

2. **Background:** There are two types of operations that can be executed by the OB contract to supply tokens as the action on receiving the message from AMB contract: unlock tokens previously locked by users, mint new tokens. These operations can be triggered by calling one of the following methods: deployAndHandleBridgedTokens, handleBridgedTokens, handleNativeTokens, deployAndHandleBridgedTokensAndCall, handleBridgedTokensAndCall and handleNativeTokensAndCall

   Claim: No other ways to mint tokens which contracts are deployed by the OB Token Fabric. No other ways to unlock tokens transferred to the OB contract the methods listed in the claim #1. Only the AMB contract is authorised to call the OB contracts method listed above.

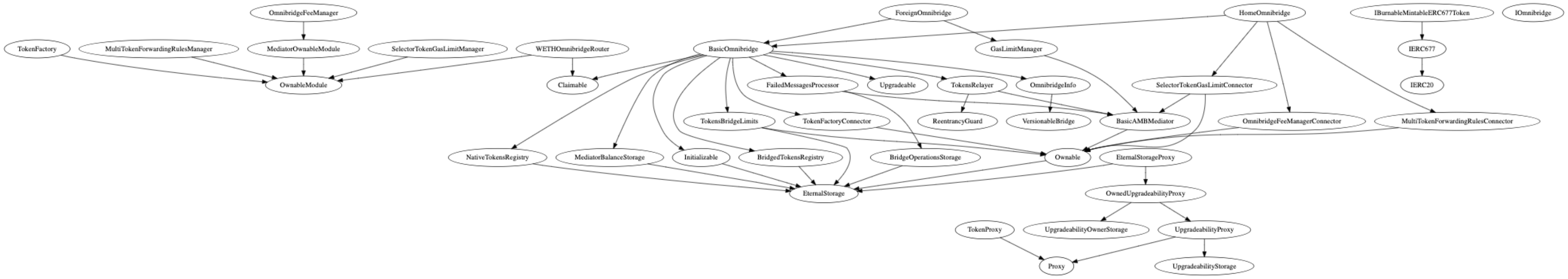
3. **Background:** The user is able to notify a recipient of tokens after their bridging by calling the method onTokenBridged of the contract-recipient. In order to request the OB contract to construct the corresponding message, the user must call either the relayTokensAndCall method of the OB contract or the transferAndCall method of the ERC677 token with the corresponding content of the field. The OB contract will use deployAndHandleBridgedTokensAndCall , handleBridgedTokensAndCall and handleNativeTokensAndCall methods in these both cases to prepare the message before passing it to AMB contracts.
   Claim: The OB contract executes the onTokenBridged method is safe manner so if the contract-recipient accidentally or purposely fails it does not affect the bridging operation: the tokens will be transferred to the recipient anyway. The OB logic is composed as so there is no way to build a request from the contract-recipient during the execution of onTokenBridged to force the OB contracts to mint/unlock extra tokens, execute another unauthorised action in the OB contract.

4. **Background:** The finalisation of the bridging process is to pass the control to the token contract by calling its transfer method. The OB contract behaves equally with all tokens so there could be cases when the token contract fails the transfer execution. It leads to the situation when the user transferred the tokens to the OB contracts on one side of the bridge, but they didn't appear on another side of the bridge. In this case, if the transfer operation fails, the user is able to invoke the requestFailedMessageFix method which will initiate the process of unlocking unbridged tokens on that side of the bridge where the initial request was originated.
   Claim: The pair of the methods requestFailedMessageFix and fixFailedMessage all ows to operate only with failed bridging operations. As part of the recovery operation, it is not possible to unlock/mint more tokens than was initially requested to be bridged. It is not possible to execute several times the recovery operation for the same failed bridge request. The AMB contract is only authorised to call the fixFailedMessage method.

5. **Background:** There is list of actions changing behaviour of OB, like: upgrade the OB implementation contract, update the addresses of the OB contract on another side of the bridge, the bridge contract, the token implementation contract, the token factory contract, managing the transfer fees and limits, fix the imbalance of the bridge, claim tokens transferred incorrectly.
   Claim: These actions can be executed only by accounts authorised in advance.

6. Overall smart contract security and business logic needs to be checked

The main goal of this audit was to verify these claims. The auditors can provide additional feedback on the code upon the client’s request.



5.1 Manual and Automated Vulnerability Test

CRITICAL ISSUES

During the audit, Chainsulting‘s experts found **no Critical issues** in the code of the smart contract.

HIGH ISSUES

During the audit, Chainsulting’s experts found **no High issues** in the code of the smart contract.

MEDIUM ISSUES

During the audit, Chainsulting’s experts found **no Medium issues** in the code of the smart contract.

LOW ISSUES

5.1.1 Overall require checks
Severity: LOW
Status: Acknowledge
https://github.com/poanetwork/omnibridge/releases/tag/1.0.0
File(s) affected: all

Attack / Description	Code Snippet	Result/Recommendation
Require checks provide a parameter for error messages, to display the reason of failing transactions.	NA	Add error messages to require checks to provide a better transparency.

5.1.2 Indexed modifier recommended
Severity: LOW
Status: Acknowledge
https://github.com/poanetwork/omnibridge/releases/tag/1.0.0
File(s) affected: OwnedUpgradeabilityProxy.sol, Ownable.sol

Attack / Description	Code Snippet	Result/Recommendation

Require checks provide a parameter for error messages, to display the reason of failing transactions.	Line 16:
	event ProxyOwnershipTransferred (address previousOwner, address newOwner);
	
	Line 18:
	event OwnershipTransferred (address previousOwner, address newOwner);
		It is recommended to use indexed addresses in events for a better filtering of logs.

INFORMATIONAL ISSUES

5.1.3 Typo error
Severity: INFORMATIONAL
Status: Fixed
https://github.com/poanetwork/omnibridge/pull/45
File(s) affected: OwnedUpgradeabilityProxy.sol

Attack / Description	Code Snippet	Result/Recommendation
Misleading comment due to typo.	Line 57: represents the msg.data to bet sent in the low level call	Use „be sent“ instead of „bet sent“.

5.1.4 Typo error

Severity: INFORMATIONAL
Status: Fixed

https://github.com/poanetwork/omnibridge/pull/45

File(s) affected: UpgradeabilityProxy.sol

Attack / Description	Code Snippet	Result/Recommendation
Typo error	Line 38: „… greater than the privios one …“	Use „previous“ instead of „privios“.

5.1.5 Hardcoded address

Severity: INFORMATIONAL

Status: Acknowledge

File(s) affected: ForeignOmnibridge.sol

Attack / Description	Code Snippet	Result/Recommendation
The xDai stake token of the ethereum mainet is hardcoded in code.	Line 133: `if (_token == address(0x0Ae055097C6d159879521C384F1D2123D1f195e6) && balance < _value)`	It is required to check the address. Also, it is required to check the code of the called contract for vulnerabilities. https://etherscan.io/address/0x0Ae055097C6d159879521C384F1D2123D1f195e6 Consider adding a clear @param about the address and link to etherscan.

6. Test Deployment

Deployment on Home Network
For deployment on home network, we used sokol from POA.

6.0.1 Deploying Bridge Mediator storage

Tx: https://blockscout.com/poa/sokol/tx/0xf106526e43fe57fee4dff9ad1df68016a76ad5f0d3175a4377824f3f697e1cd3
Contract: https://blockscout.com/poa/sokol/address/0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5

6.0.2 Deploying new ERC677 Token image

Tx: https://blockscout.com/poa/sokol/tx/0x88f51f1802220e03fa3aee6a8c35bdae5d80fad379176b64c7a3462e2f7df764
Contract: https://blockscout.com/poa/sokol/address/0x5407fD280899994810c3645795b73479b5D897cd

6.0.3 Deploying new token factory

Tx: https://blockscout.com/poa/sokol/tx/0x99132ce09d8b81c3e76e4f5b69dd7b0e392d013a02c9fe255c9febf1b1dac139
Contract: https://blockscout.com/poa/sokol/address/0x0Faf2D217adF42c061fF11A8d5fA8463C3980568

6.0.4 Deploying gas limit manager contract

Following parameters are used for the deployment:
HOME_AMB_BRIDGE: 0xFe446bEF1DbF7AFE24E81e05BC8B271C1BA9a560
OWNER: 0x28d12e63Bf7d8Ed75F12fC079c316aD9A236B358
HOME_MEDIATOR_REQUEST_GAS_LIMIT: 2000000

Tx: https://blockscout.com/poa/sokol/tx/0xb05ebc96953b02e268d38ffbb12c64027e007070162c001474da008dc36f3fd3
Contract: https://blockscout.com/poa/sokol/address/0x5020d6a9Bf4c2259Ee430d6E3dA16617497fDEd1

6.0.5 Deploying Bridge Mediator implementation
Tx: https://blockscout.com/poa/sokol/tx/0x97122262041cc48d9b4666206e26c819746b7241f6bb78699a4cccfbbd14537d
Contract: https://blockscout.com/poa/sokol/address/0x2EE209DDdD921adB7c891660675b05AF1C7DCDf3

6.0.6 Hooking up Mediator storage to Mediator implementation
Tx: https://blockscout.com/poa/sokol/tx/0x761280ca8d3e99950513cba3bce7613ec1bc8068e0237c28cf074728ae74a659

Deployment on Foreign Network
For deployment on foreign network, we used kovan testnet.

6.0.7 Deploying Bridge Mediator storage
Tx: https://kovan.etherscan.io/tx/0x862dbf3449d2e344c7df6dff66a00b85d65f9577fc1773d74a911f39f102dbd6
Contract: https://kovan.etherscan.io/address/0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5

6.0.8 Deploying new ERC677 token image
Tx: https://kovan.etherscan.io/tx/0xf0f4bf8f66b65fba8dd3d0465e5ec97309983d11d2402c937b7e049069faec90
Contract: https://kovan.etherscan.io/address/0x5407fD280899994810c3645795b73479b5D897cd

6.0.9 Deploying new token factory
Tx: https://kovan.etherscan.io/tx/0x8468acab1c43d78720bf9826a5a8e442bc49cfee897efca317bde73f906cfd84
Contract: https://kovan.etherscan.io/address/0x0Faf2D217adF42c061fF11A8d5fA8463C3980568

6.0.10 Deploying Bridge Mediator implementation
Tx: https://kovan.etherscan.io/tx/0xfe0d57c156cfd72f9b22a23d16e93a54ab10e10864fb53b0bda498dde0fafffc
Contract: https://kovan.etherscan.io/address/0x5020d6a9Bf4c2259Ee430d6E3dA16617497fDEd1

6.0.11 Hooking up Mediator storage to Mediator implementation
Tx: https://kovan.etherscan.io/tx/0x0192cfe0ebc3e0e1bf8678d86bf253e91d45c4f896ad0cb8eeb58c4632445f91

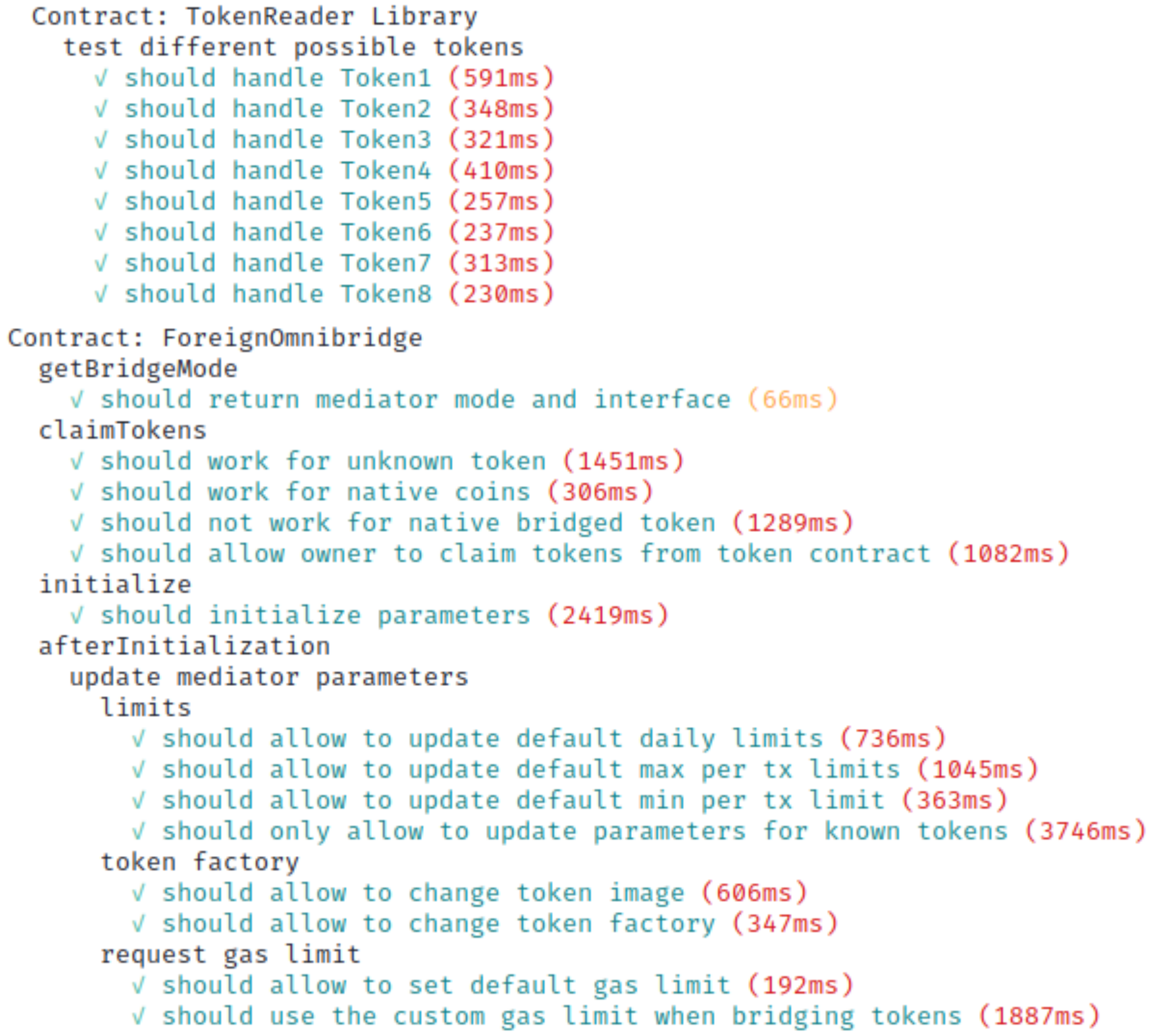
6.0.12 Initializing Home Bridge Mediator

The Home Bridge Mediator is initialized with the following parameters:
AMB contract: 0xFe446bEF1DbF7AFE24E81e05BC8B271C1BA9a560,
Mediator contract: 0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,,
DAILY_LIMIT: 30000000000000000000000000 which is 30000000 in eth,
MAX_AMOUNT_PER_TX: 1500000000000000000000000 which is 1500000 in eth,
MIN_AMOUNT_PER_TX: 500000000000000000 which is 0.5 in eth,
EXECUTION_DAILY_LIMIT: 15000000000000000000000000 which is 15000000 in eth,
EXECUTION_MAX_AMOUNT_PER_TX: 750000000000000000000000 which is 750000 in eth,
OWNER: 0x28d12e63Bf7d8Ed75F12fC079c316aD9A236B358,
TOKEN_FACTORY: 0x0Faf2D217adF42c061fF11A8d5fA8463C3980568,
FEE_MANAGER: 0x0000000000000000000000000000000000000000,
GAS_LIMIT_MANAGER: 0x5020d6a9Bf4c2259Ee430d6E3dA16617497fDEd1,
FORWARDING_RULES_MANAGER: 0x0000000000000000000000000000000000000000,
Tx: https://blockscout.com/poa/sokol/tx/0xef05b4f0c4d9f03702e6840922eb10526a6e9b9de60de359f4548de1c080250d

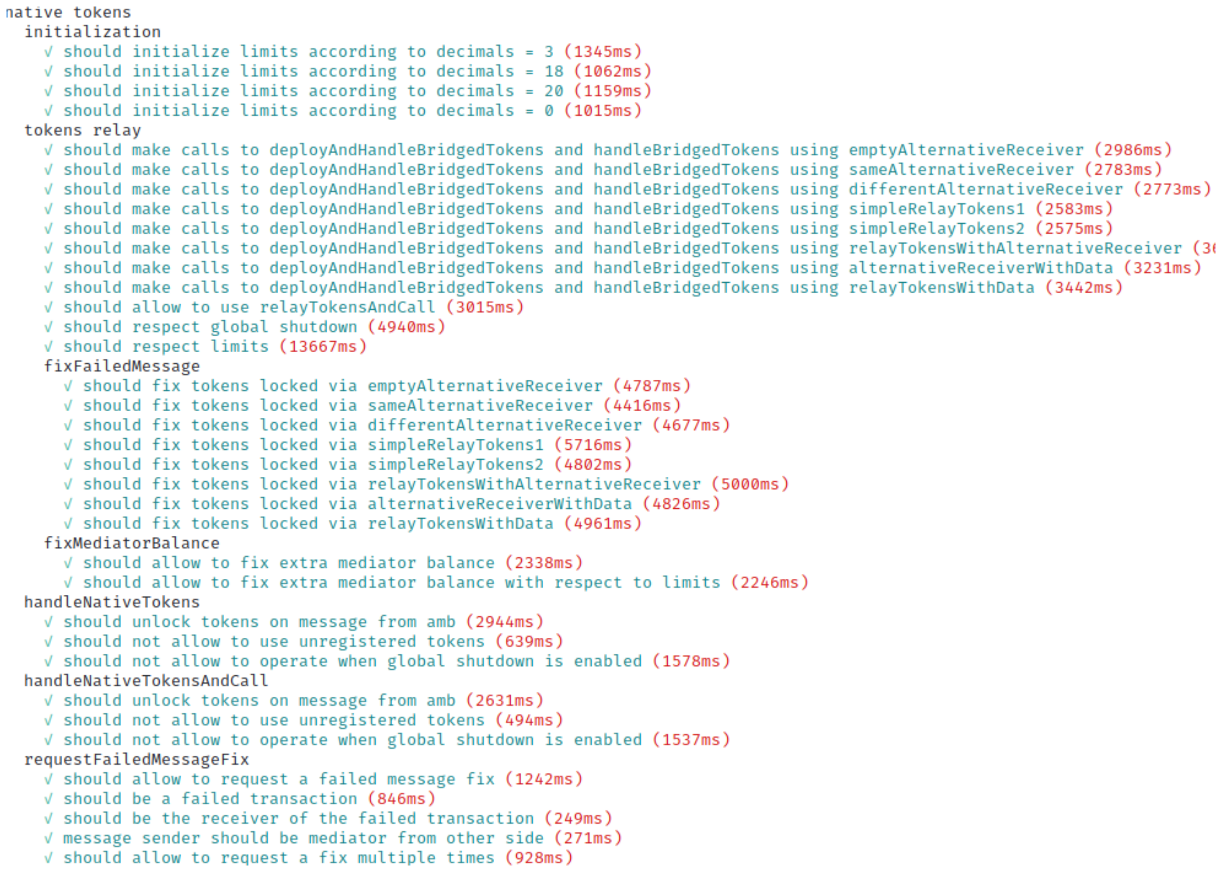
6.0.13 Initializing Foreign Bridge Mediator

The Foreign Bridge Mediator is initilized with the following parameters:
AMB contract: 0xFe446bEF1DbF7AFE24E81e05BC8B271C1BA9a560,
Mediator contract: 0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,
DAILY_LIMIT: 15000000000000000000000000 which is 15000000 in eth,
MAX_AMOUNT_PER_TX: 750000000000000000000000 which is 750000 in eth,
MIN_AMOUNT_PER_TX: 500000000000000000 which is 0.5 in eth,
EXECUTION_DAILY_LIMIT: 30000000000000000000000000 which is 30000000 in eth,
EXECUTION_MAX_AMOUNT_PER_TX: 1500000000000000000000000 which is 1500000 in eth,
MEDIATOR_REQUEST_GAS_LIMIT: 2000000,
OWNER: 0x28d12e63Bf7d8Ed75F12fC079c316aD9A236B358,
TOKEN_FACTORY: 0x0Faf2D217adF42c061fF11A8d5fA8463C3980568
Tx: https://kovan.etherscan.io/tx/0x69621ca5f56adadb67a8073f6e89ee5da9fa6162685eebf49ed23627707a2f4b

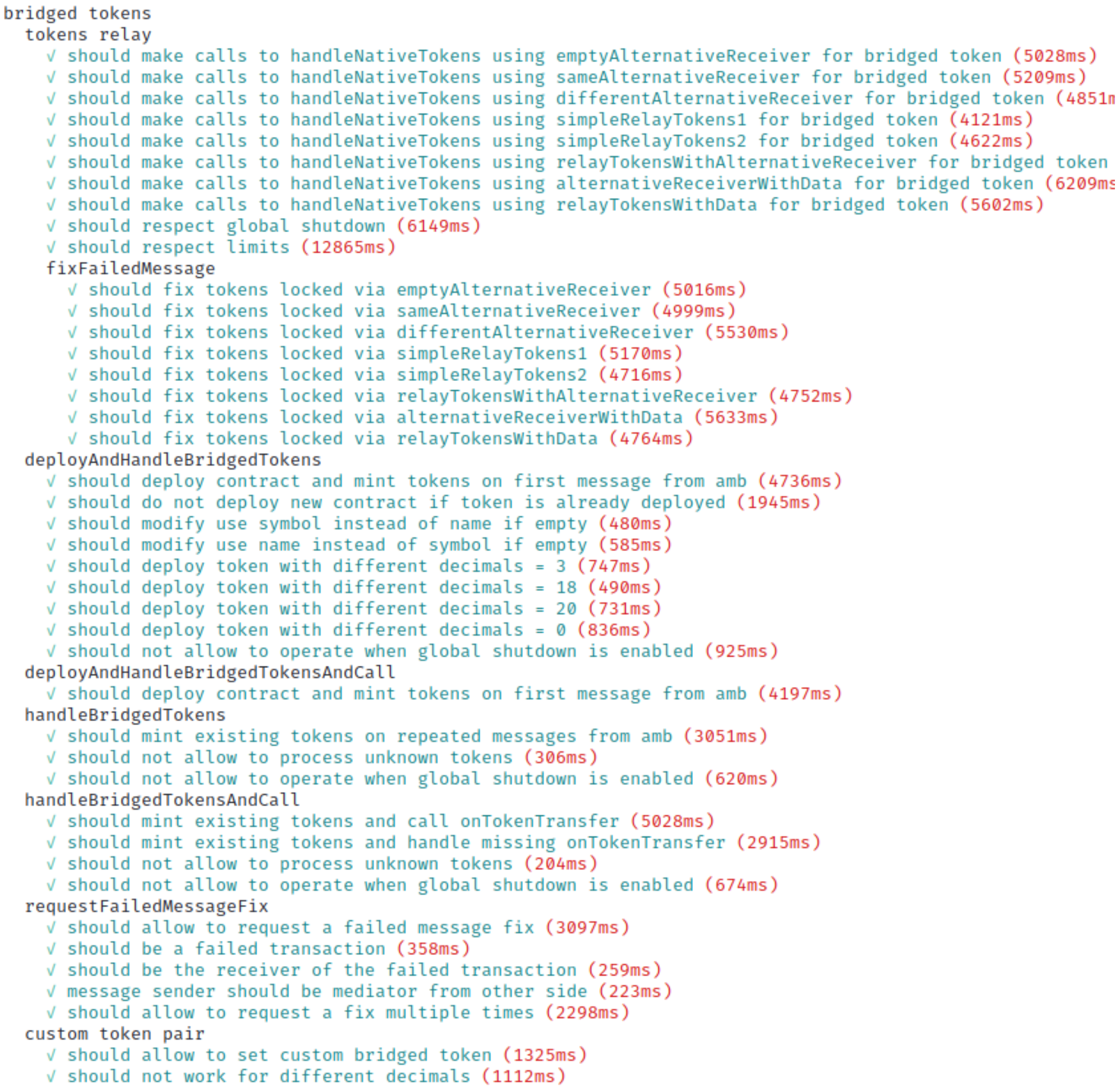
6.1 Unit Test



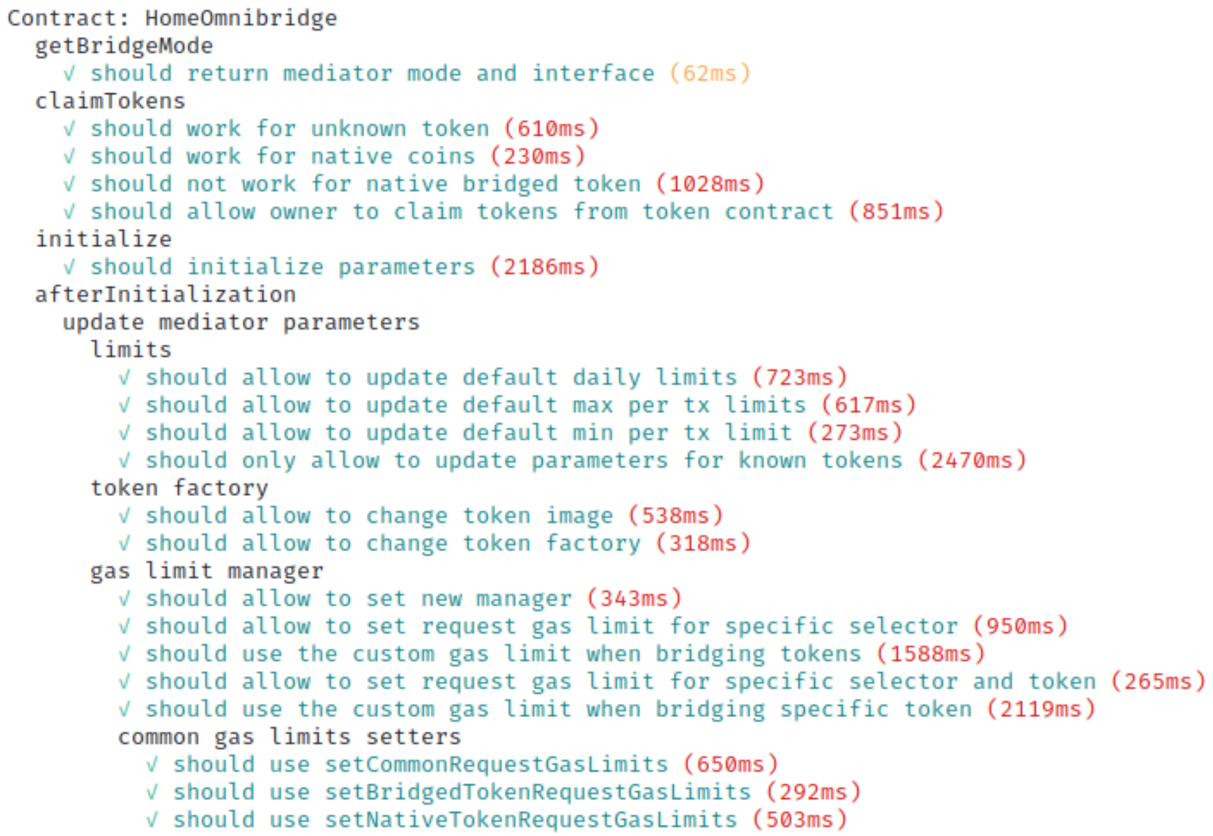
```
  Contract: TokenReader Library
    test different possible tokens
      √ should handle Token1 (591ms)
      √ should handle Token2 (348ms)
      √ should handle Token3 (321ms)
      √ should handle Token4 (410ms)
      √ should handle Token5 (257ms)
      √ should handle Token6 (237ms)
      √ should handle Token7 (313ms)
      √ should handle Token8 (230ms)

Contract: ForeignOmnibridge
  getBridgeMode
    √ should return mediator mode and interface (66ms)
  claimTokens
    √ should work for unknown token (1451ms)
    √ should work for native coins (306ms)
    √ should not work for native bridged token (1289ms)
    √ should allow owner to claim tokens from token contract (1082ms)
  initialize
    √ should initialize parameters (2419ms)
  afterInitialization
    update mediator parameters
      limits
        √ should allow to update default daily limits (736ms)
        √ should allow to update default max per tx limits (1045ms)
        √ should allow to update default min per tx limit (363ms)
        √ should only allow to update parameters for known tokens (3746ms)
      token factory
        √ should allow to change token image (606ms)
        √ should allow to change token factory (347ms)
      request gas limit
        √ should allow to set default gas limit (192ms)
        √ should use the custom gas limit when bridging tokens (1887ms)
```



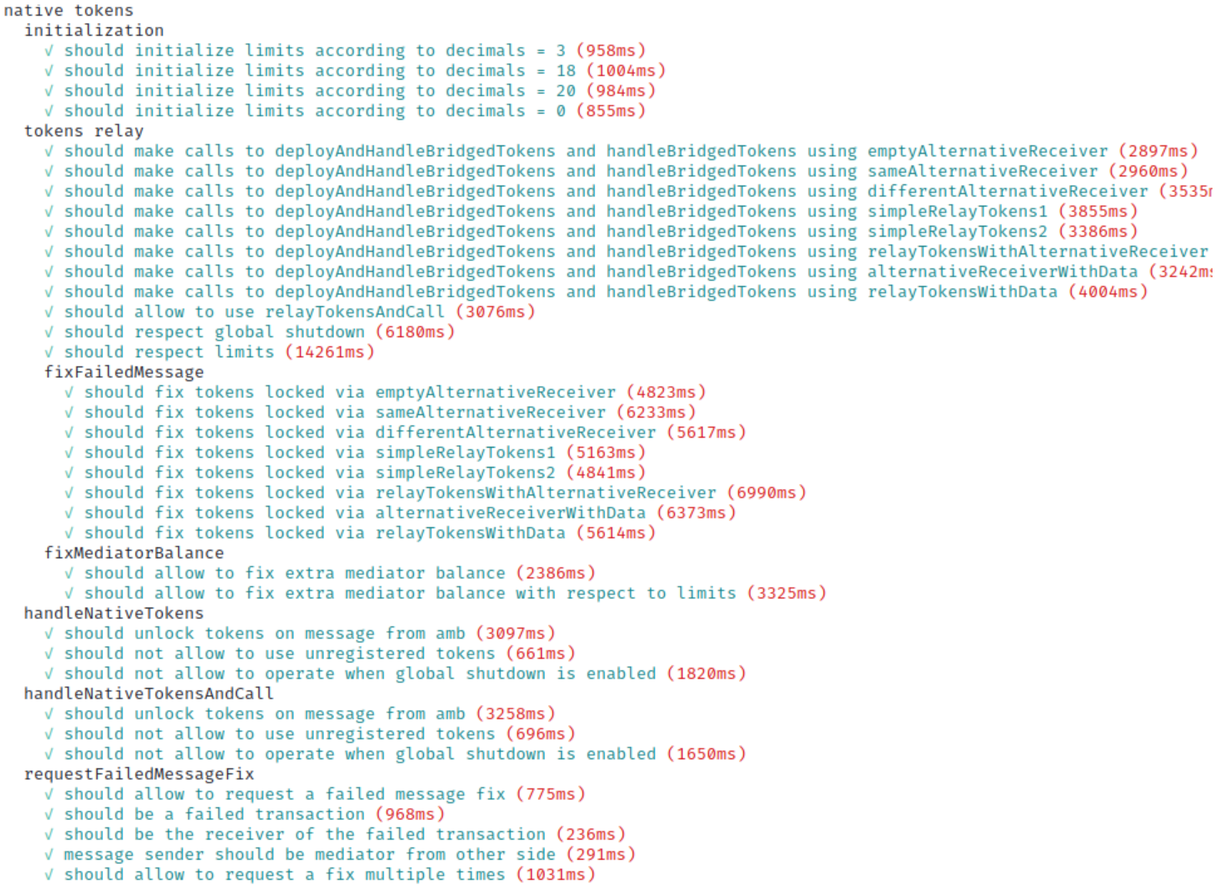
```
native tokens
  initialization
    √ should initialize limits according to decimals = 3 (1345ms)
    √ should initialize limits according to decimals = 18 (1062ms)
    √ should initialize limits according to decimals = 20 (1159ms)
    √ should initialize limits according to decimals = 0 (1015ms)
  tokens relay
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using emptyAlternativeReceiver (2986ms)
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using sameAlternativeReceiver (2783ms)
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using differentAlternativeReceiver (2773ms)
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using simpleRelayTokens1 (2583ms)
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using simpleRelayTokens2 (2575ms)
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using relayTokensWithAlternativeReceiver (3
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using alternativeReceiverWithData (3231ms)
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using relayTokensWithData (3442ms)
    √ should allow to use relayTokensAndCall (3015ms)
    √ should respect global shutdown (4940ms)
    √ should respect limits (13667ms)
    fixFailedMessage
      √ should fix tokens locked via emptyAlternativeReceiver (4787ms)
      √ should fix tokens locked via sameAlternativeReceiver (4416ms)
      √ should fix tokens locked via differentAlternativeReceiver (4677ms)
      √ should fix tokens locked via simpleRelayTokens1 (5716ms)
      √ should fix tokens locked via simpleRelayTokens2 (4802ms)
      √ should fix tokens locked via relayTokensWithAlternativeReceiver (5000ms)
      √ should fix tokens locked via alternativeReceiverWithData (4826ms)
      √ should fix tokens locked via relayTokensWithData (4961ms)
    fixMediatorBalance
      √ should allow to fix extra mediator balance (2338ms)
      √ should allow to fix extra mediator balance with respect to limits (2246ms)
  handleNativeTokens
    √ should unlock tokens on message from amb (2944ms)
    √ should not allow to use unregistered tokens (639ms)
    √ should not allow to operate when global shutdown is enabled (1578ms)
  handleNativeTokensAndCall
    √ should unlock tokens on message from amb (2631ms)
    √ should not allow to use unregistered tokens (494ms)
    √ should not allow to operate when global shutdown is enabled (1537ms)
  requestFailedMessageFix
    √ should allow to request a failed message fix (1242ms)
    √ should be a failed transaction (846ms)
    √ should be the receiver of the failed transaction (249ms)
    √ message sender should be mediator from other side (271ms)
    √ should allow to request a fix multiple times (928ms)
```



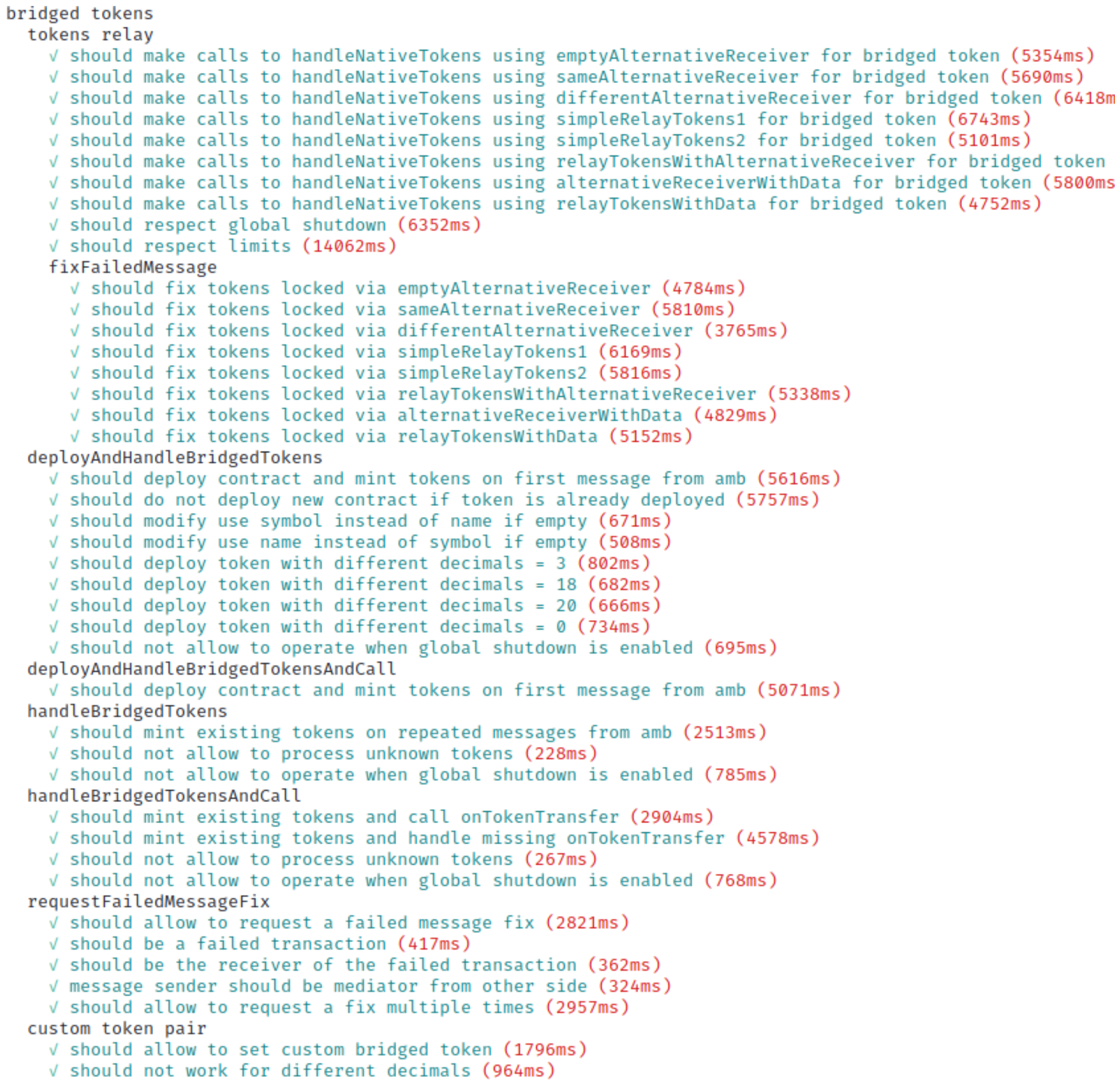
```
bridged tokens
  tokens relay
    √ should make calls to handleNativeTokens using emptyAlternativeReceiver for bridged token (5028ms)
    √ should make calls to handleNativeTokens using sameAlternativeReceiver for bridged token (5209ms)
    √ should make calls to handleNativeTokens using differentAlternativeReceiver for bridged token (4851m
    √ should make calls to handleNativeTokens using simpleRelayTokens1 for bridged token (4121ms)
    √ should make calls to handleNativeTokens using simpleRelayTokens2 for bridged token (4622ms)
    √ should make calls to handleNativeTokens using relayTokensWithAlternativeReceiver for bridged token
    √ should make calls to handleNativeTokens using alternativeReceiverWithData for bridged token (6209ms
    √ should make calls to handleNativeTokens using relayTokensWithData for bridged token (5602ms)
    √ should respect global shutdown (6149ms)
    √ should respect limits (12865ms)
    fixFailedMessage
      √ should fix tokens locked via emptyAlternativeReceiver (5016ms)
      √ should fix tokens locked via sameAlternativeReceiver (4999ms)
      √ should fix tokens locked via differentAlternativeReceiver (5530ms)
      √ should fix tokens locked via simpleRelayTokens1 (5170ms)
      √ should fix tokens locked via simpleRelayTokens2 (4716ms)
      √ should fix tokens locked via relayTokensWithAlternativeReceiver (4752ms)
      √ should fix tokens locked via alternativeReceiverWithData (5633ms)
      √ should fix tokens locked via relayTokensWithData (4764ms)
  deployAndHandleBridgedTokens
    √ should deploy contract and mint tokens on first message from amb (4736ms)
    √ should do not deploy new contract if token is already deployed (1945ms)
    √ should modify use symbol instead of name if empty (480ms)
    √ should modify use name instead of symbol if empty (585ms)
    √ should deploy token with different decimals = 3 (747ms)
    √ should deploy token with different decimals = 18 (490ms)
    √ should deploy token with different decimals = 20 (731ms)
    √ should deploy token with different decimals = 0 (836ms)
    √ should not allow to operate when global shutdown is enabled (925ms)
  deployAndHandleBridgedTokensAndCall
    √ should deploy contract and mint tokens on first message from amb (4197ms)
  handleBridgedTokens
    √ should mint existing tokens on repeated messages from amb (3051ms)
    √ should not allow to process unknown tokens (306ms)
    √ should not allow to operate when global shutdown is enabled (620ms)
  handleBridgedTokensAndCall
    √ should mint existing tokens and call onTokenTransfer (5028ms)
    √ should mint existing tokens and handle missing onTokenTransfer (2915ms)
    √ should not allow to process unknown tokens (204ms)
    √ should not allow to operate when global shutdown is enabled (674ms)
  requestFailedMessageFix
    √ should allow to request a failed message fix (3097ms)
    √ should be a failed transaction (358ms)
    √ should be the receiver of the failed transaction (259ms)
    √ message sender should be mediator from other side (223ms)
    √ should allow to request a fix multiple times (2298ms)
  custom token pair
    √ should allow to set custom bridged token (1325ms)
    √ should not work for different decimals (1112ms)
```



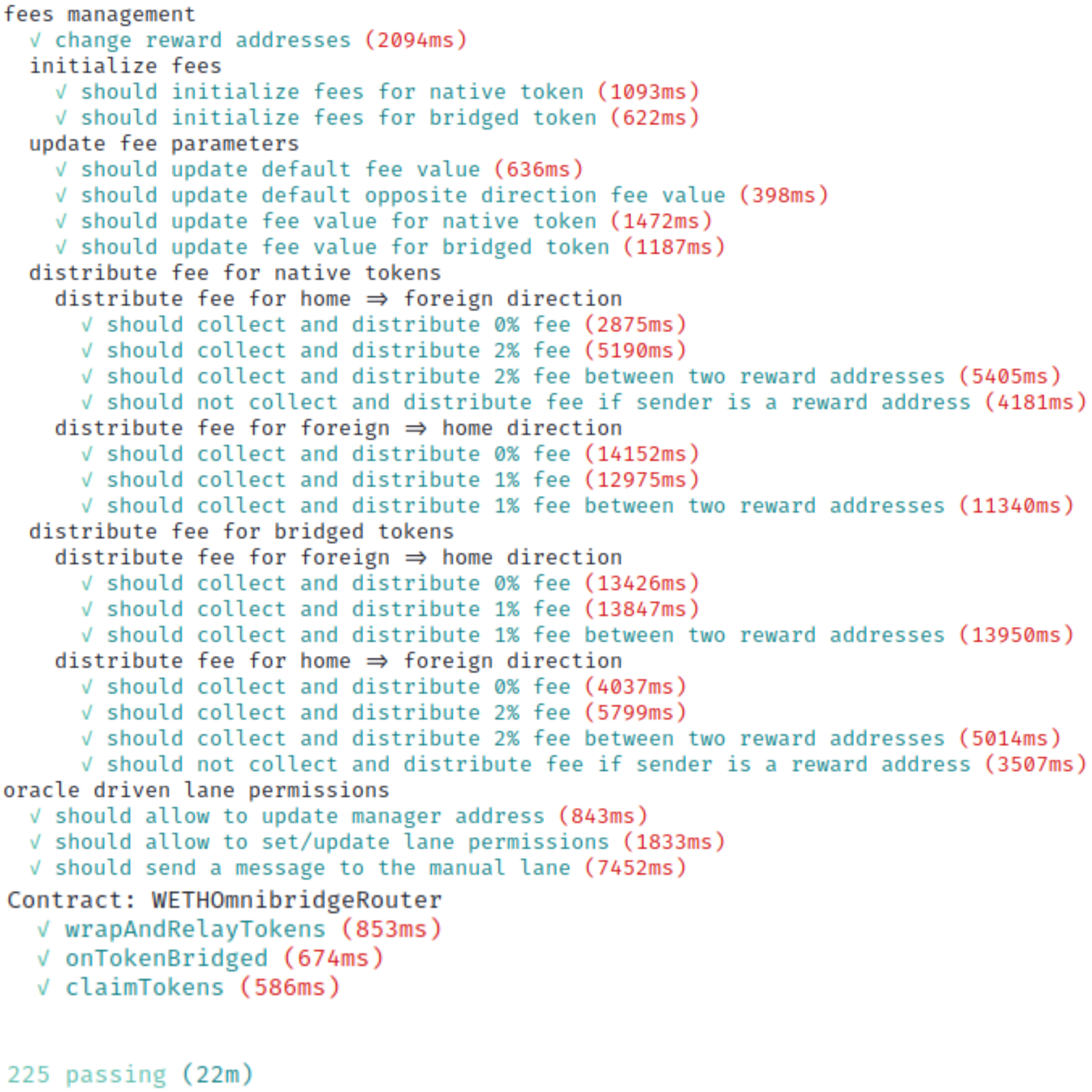
```
Contract: HomeOmnibridge
  getBridgeMode
    √ should return mediator mode and interface (62ms)
  claimTokens
    √ should work for unknown token (610ms)
    √ should work for native coins (230ms)
    √ should not work for native bridged token (1028ms)
    √ should allow owner to claim tokens from token contract (851ms)
  initialize
    √ should initialize parameters (2186ms)
  afterInitialization
    update mediator parameters
      limits
        √ should allow to update default daily limits (723ms)
        √ should allow to update default max per tx limits (617ms)
        √ should allow to update default min per tx limit (273ms)
        √ should only allow to update parameters for known tokens (2470ms)
      token factory
        √ should allow to change token image (538ms)
        √ should allow to change token factory (318ms)
      gas limit manager
        √ should allow to set new manager (343ms)
        √ should allow to set request gas limit for specific selector (950ms)
        √ should use the custom gas limit when bridging tokens (1588ms)
        √ should allow to set request gas limit for specific selector and token (265ms)
        √ should use the custom gas limit when bridging specific token (2119ms)
        common gas limits setters
          √ should use setCommonRequestGasLimits (650ms)
          √ should use setBridgedTokenRequestGasLimits (292ms)
          √ should use setNativeTokenRequestGasLimits (503ms)
```



```
native tokens
  initialization
    √ should initialize limits according to decimals = 3 (958ms)
    √ should initialize limits according to decimals = 18 (1004ms)
    √ should initialize limits according to decimals = 20 (984ms)
    √ should initialize limits according to decimals = 0 (855ms)
  tokens relay
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using emptyAlternativeReceiver (2897ms)
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using sameAlternativeReceiver (2960ms)
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using differentAlternativeReceiver (3535
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using simpleRelayTokens1 (3855ms)
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using simpleRelayTokens2 (3386ms)
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using relayTokensWithAlternativeReceiver
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using alternativeReceiverWithData (3242m
    √ should make calls to deployAndHandleBridgedTokens and handleBridgedTokens using relayTokensWithData (4004ms)
    √ should allow to use relayTokensAndCall (3076ms)
    √ should respect global shutdown (6180ms)
    √ should respect limits (14261ms)
    fixFailedMessage
      √ should fix tokens locked via emptyAlternativeReceiver (4823ms)
      √ should fix tokens locked via sameAlternativeReceiver (6233ms)
      √ should fix tokens locked via differentAlternativeReceiver (5617ms)
      √ should fix tokens locked via simpleRelayTokens1 (5163ms)
      √ should fix tokens locked via simpleRelayTokens2 (4841ms)
      √ should fix tokens locked via relayTokensWithAlternativeReceiver (6990ms)
      √ should fix tokens locked via alternativeReceiverWithData (6373ms)
      √ should fix tokens locked via relayTokensWithData (5614ms)
    fixMediatorBalance
      √ should allow to fix extra mediator balance (2386ms)
      √ should allow to fix extra mediator balance with respect to limits (3325ms)
  handleNativeTokens
    √ should unlock tokens on message from amb (3097ms)
    √ should not allow to use unregistered tokens (661ms)
    √ should not allow to operate when global shutdown is enabled (1820ms)
  handleNativeTokensAndCall
    √ should unlock tokens on message from amb (3258ms)
    √ should not allow to use unregistered tokens (696ms)
    √ should not allow to operate when global shutdown is enabled (1650ms)
  requestFailedMessageFix
    √ should allow to request a failed message fix (775ms)
    √ should be a failed transaction (968ms)
    √ should be the receiver of the failed transaction (236ms)
    √ message sender should be mediator from other side (291ms)
    √ should allow to request a fix multiple times (1031ms)
```



```
bridged tokens
  tokens relay
    √ should make calls to handleNativeTokens using emptyAlternativeReceiver for bridged token (5354ms)
    √ should make calls to handleNativeTokens using sameAlternativeReceiver for bridged token (5690ms)
    √ should make calls to handleNativeTokens using differentAlternativeReceiver for bridged token (6418m
    √ should make calls to handleNativeTokens using simpleRelayTokens1 for bridged token (6743ms)
    √ should make calls to handleNativeTokens using simpleRelayTokens2 for bridged token (5101ms)
    √ should make calls to handleNativeTokens using relayTokensWithAlternativeReceiver for bridged token
    √ should make calls to handleNativeTokens using alternativeReceiverWithData for bridged token (5800ms
    √ should make calls to handleNativeTokens using relayTokensWithData for bridged token (4752ms)
    √ should respect global shutdown (6352ms)
    √ should respect limits (14062ms)
    fixFailedMessage
      √ should fix tokens locked via emptyAlternativeReceiver (4784ms)
      √ should fix tokens locked via sameAlternativeReceiver (5810ms)
      √ should fix tokens locked via differentAlternativeReceiver (3765ms)
      √ should fix tokens locked via simpleRelayTokens1 (6169ms)
      √ should fix tokens locked via simpleRelayTokens2 (5816ms)
      √ should fix tokens locked via relayTokensWithAlternativeReceiver (5338ms)
      √ should fix tokens locked via alternativeReceiverWithData (4829ms)
      √ should fix tokens locked via relayTokensWithData (5152ms)
  deployAndHandleBridgedTokens
    √ should deploy contract and mint tokens on first message from amb (5616ms)
    √ should do not deploy new contract if token is already deployed (5757ms)
    √ should modify use symbol instead of name if empty (671ms)
    √ should modify use name instead of symbol if empty (508ms)
    √ should deploy token with different decimals = 3 (802ms)
    √ should deploy token with different decimals = 18 (682ms)
    √ should deploy token with different decimals = 20 (666ms)
    √ should deploy token with different decimals = 0 (734ms)
    √ should not allow to operate when global shutdown is enabled (695ms)
  deployAndHandleBridgedTokensAndCall
    √ should deploy contract and mint tokens on first message from amb (5071ms)
  handleBridgedTokens
    √ should mint existing tokens on repeated messages from amb (2513ms)
    √ should not allow to process unknown tokens (228ms)
    √ should not allow to operate when global shutdown is enabled (785ms)
  handleBridgedTokensAndCall
    √ should mint existing tokens and call onTokenTransfer (2904ms)
    √ should mint existing tokens and handle missing onTokenTransfer (4578ms)
    √ should not allow to process unknown tokens (267ms)
    √ should not allow to operate when global shutdown is enabled (768ms)
  requestFailedMessageFix
    √ should allow to request a failed message fix (2821ms)
    √ should be a failed transaction (417ms)
    √ should be the receiver of the failed transaction (362ms)
    √ message sender should be mediator from other side (324ms)
    √ should allow to request a fix multiple times (2957ms)
  custom token pair
    √ should allow to set custom bridged token (1796ms)
    √ should not work for different decimals (964ms)
```



```
fees management
  √ change reward addresses (2094ms)
  initialize fees
    √ should initialize fees for native token (1093ms)
    √ should initialize fees for bridged token (622ms)
  update fee parameters
    √ should update default fee value (636ms)
    √ should update default opposite direction fee value (398ms)
    √ should update fee value for native token (1472ms)
    √ should update fee value for bridged token (1187ms)
  distribute fee for native tokens
    distribute fee for home ⇒ foreign direction
      √ should collect and distribute 0% fee (2875ms)
      √ should collect and distribute 2% fee (5190ms)
      √ should collect and distribute 2% fee between two reward addresses (5405ms)
      √ should not collect and distribute fee if sender is a reward address (4181ms)
    distribute fee for foreign ⇒ home direction
      √ should collect and distribute 0% fee (14152ms)
      √ should collect and distribute 1% fee (12975ms)
      √ should collect and distribute 1% fee between two reward addresses (11340ms)
  distribute fee for bridged tokens
    distribute fee for foreign ⇒ home direction
      √ should collect and distribute 0% fee (13426ms)
      √ should collect and distribute 1% fee (13847ms)
      √ should collect and distribute 1% fee between two reward addresses (13950ms)
    distribute fee for home ⇒ foreign direction
      √ should collect and distribute 0% fee (4037ms)
      √ should collect and distribute 2% fee (5799ms)
      √ should collect and distribute 2% fee between two reward addresses (5014ms)
      √ should not collect and distribute fee if sender is a reward address (3507ms)
oracle driven lane permissions
  √ should allow to update manager address (843ms)
  √ should allow to set/update lane permissions (1833ms)
  √ should send a message to the manual lane (7452ms)

Contract: WETHOmnibridgeRouter
  √ wrapAndRelayTokens (853ms)
  √ onTokenBridged (674ms)
  √ claimTokens (586ms)


225 passing (22m)
```

6.2 E2E Test

```
Initializing test environment
Import accounts
Initializing mediators contracts
Initializing AMB contracts
Fetching fee values
Home fee: 0%
Foreign fee: 0%
Initializing tokens
Deploying test Home token
pending tx: 0xde393f1addb1cfff572a719a7f311900d3e842f1a706ddb20c815e908153c31c
Deployed token 0x6acAB06915A93DD5dfBF10dA04D7175b5677D587
Minting 1000 tokens to the 0x16016b0ACd9192eF51Af129aB75d110178c79C14
pending tx: 0xa251e01f61b1f68ed11afd43225a9f90285b743140d24b2ebdb8a9168d8f399f
Deploying test Foreign token
pending tx: 0xc077a1875b36c5b19077072552c7f8c05c9882008e4c2e14194669d82ac5339a
Deployed token 0x6acAB06915A93DD5dfBF10dA04D7175b5677D587
Minting 1000 tokens to the 0x16016b0ACd9192eF51Af129aB75d110178c79C14
pending tx: 0x5379420a27afdf8501dd2dc5af697328dd698fe120f3def0ac1082ab92a6544a
Deploying test Home claimable token
pending tx: 0x7342cbb2e142e9d691877fa7006de9fefce00f822095adb7532b805451177718
Deployed token 0x5C59Cd4f18c4eBf3Ac603D4D5804E0dfDFe5e73A
Minting 1000 tokens to the 0x16016b0ACd9192eF51Af129aB75d110178c79C14
pending tx: 0x970218953cb1d7722a2b54520b6d2d64dc1bac8b305fc302588ce3cd86380b71
Deploying test Foreign claimable token
pending tx: 0x2d608181f45ba3558a0fa860d36c2ac6ba242434fd7a480b41224754d27f55f3
Deployed token 0x5C59Cd4f18c4eBf3Ac603D4D5804E0dfDFe5e73A
Minting 1000 tokens to the 0x16016b0ACd9192eF51Af129aB75d110178c79C14
pending tx: 0xd43e4ded39bf17321588bb37f2622c7f2aa7ff7637e290655893cd76ce7befc9
pending tx: 0xf4070866c5cc6a3ae9c55edc4fb5a4837666bc9298e9480a3bc64606941f67bb
Deployed token receiver 0x5Dcec429e1a46d8A8517DbA74E1733F763FB556F
pending tx: 0xdd98b5e9b65ad61e0cc39d8d7bedd167044328b31d5891f568f3cbf84c4240fd
```

```
Deployed token receiver 0x5Dcec429e1a46d8A8517DbA74E1733F763FB556F
Fetching block numbers
Initializing WETH stack
pending tx: 0x47e12cda7a6e324403234cc260d73daabb4c5d502d1082229eef9afd0fedea40
Deployed WETH token 0x95474Aafbc51E1Db090Acea952dD0E1E2a69555f
pending tx: 0x05d0bcf32027113628068d7b7ff2d470aa4905d54758ce6cee122d7f5f2673c1
Deployed WETHOmnibridgeRouter token 0x7Fa92e2419F6E5DC51aA2399b33b06d958De218E

Running scenario 1/13 - Claiming of foreign tokens
Claiming Foreign erc20 tokens
Sending 10 tokens to the Foreign Mediator
pending tx: 0x5495912431ffc0632b7b7a85b2e5226ac8d365f569fc5950a8dd99b7613128c0
Sending claim request to the Foreign Mediator
pending tx: 0x5b1d45b4236d844b53b25f130a84cb2b554d21b38f787e493fe54e5afe74cefe
Checking if transaction has the required Transfer(0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, any)
OK
Running scenario 2/13 - Claiming of home tokens

Claiming Home erc20 tokens
Sending 10 tokens to the Home Mediator
pending tx: 0x35eb666d442d4ecf771e209a5a9d73d5d9e5d631b43b2229cfbf750cdc2cd72d
Sending claim request to the Home Mediator
pending tx: 0x4e01b7607418f473feee023dee529d15910f3a8caf78d80925439eb3477b398b
Checking if transaction has the required Transfer(0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, any)
OK
Running scenario 3/13 - Bridging of native Foreign tokens in both directions

Bridging Native Foreign token to Home chain
Sending 10 tokens to the Foreign Mediator
pending tx: 0xdb6ed8255628b304c2ebde19d099f95f5714c852cabe215a546d382e4c3209b8
```

```
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000ec2 to be processed
Getting address of the bridged token
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 10000000000000000000)

Sending 10 more tokens to the Foreign Mediator
pending tx: 0xe29c4ab8640cead21948706a568045778610ea0f163e09c47b38f93922b6e385
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000ec3 to be processed
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 10000000000000000000)

Sending 10 bridged tokens to the Home Mediator
pending tx: 0x693dc59f2c2f594497519b6844a83394a5dd6bd73ad35b32f0fdb71710cc84e3
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de731000000000000065f to be processed
Checking if transaction has the required Transfer(0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 10000000000000000000)
OK
Running scenario 4/13 - Bridging of native Home tokens in both directions

Bridging Native Home token to Foreign chain
Sending 10 tokens to the Home Mediator
pending tx: 0x1f180452f5ce4f21935ae1286b0334a01efd3716f28eab238daa6db5450d020d
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de7310000000000000660 to be processed
Getting address of the bridged token
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 10000000000000000000)

Sending 10 more tokens to the Home Mediator
pending tx: 0xa87f3ea4426ae58921e2336e61e8f0fd25cdf987122f28281ac05206ad923948
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de7310000000000000661 to be processed
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 10000000000000000000)
```

```
Sending 10 bridged tokens to the Foreign Mediator
pending tx: 0xa17ee76340295d626137dfedc926f8ede80dbf4d799c082d9312ade53c3ae346
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b21326000000000000000ec4 to be processed
Checking if transaction has the required Transfer(0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 10000000000000000000)
OK

Running scenario 5/13 - Bridging of native Foreign tokens in both directions with alternative receiver

Bridging Native Foreign token to Home chain with alternative receiver
Sending 10 tokens to the Foreign Mediator
pending tx: 0x51ca2e00e93875f53b12457e9246d30f7b97390547802b23c391075695589790
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b21326000000000000000ec5 to be processed
Getting address of the bridged token
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x956Fb548555fEf093704482A62f18B930eB7406D, 10000000000000000000)

Sending 10 more tokens to the Foreign Mediator
pending tx: 0xb139b7965804d5218d48a1243b377e99e07a6ed4b71ea9ec5686355b392ee072
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b21326000000000000000ec6 to be processed
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x956Fb548555fEf093704482A62f18B930eB7406D, 10000000000000000000)

Sending 10 bridged tokens to the Home Mediator
pending tx: 0xf804c905b825048d77e37646a7a0608a19d48b9fba5b184c69eb3f16089dcdaf
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de7310000000000000663 to be processed
Checking if transaction has the required Transfer(0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 10000000000000000000)
OK

Running scenario 6/13 - Bridging of native Home tokens in both directions with alternative receiver
```

```
Bridging Native Home token to Foreign chain with alternative receiver
Sending 10 tokens to the Home Mediator
pending tx: 0x0b0c78d43b03ff9837aa669bd8e06e91e7e167569ef2b93e6814d7baea0fa5c5
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de7310000000000000664 to be processed
Getting address of the bridged token
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x956Fb548555fEf093704482A62f18B930eB7406D, 10000000000000000000)

Sending 10 more tokens to the Home Mediator
pending tx: 0x3daec5fed9ffb7edec3ad85c961f4f7f375566e4ce32bf973d63bfb9a8884cd2
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de7310000000000000665 to be processed
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x956Fb548555fEf093704482A62f18B930eB7406D, 10000000000000000000)

Sending 10 bridged tokens to the Foreign Mediator
pending tx: 0x8991f8c93bc84be892c1005d3599b17eaab9cf839a565d0485de411d2e9766e2
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000ec7 to be processed
Checking if transaction has the required Transfer(0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 10000000000000000000)
```

OK

Running scenario 7/13 - Fixing mediator balance of the foreign mediator

```
Fixing mediator balance of the foreign mediator
Sending 10 tokens to the Foreign Mediator
pending tx: 0xcb87864dc55bb44917b2e9df128d53c733af9cdae5af5aff2fc41f3660045a8b
Balance diff 10000000000000000000
Sending fixMediatorBalance request to the Foreign Mediator
pending tx: 0x2696a83bda56ce7c841705d5d86cfab28bdce2a2b24e69fd9448c8cf9998238e
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000ec8 to be processed
Getting address of the bridged token
```

```
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 10000000000000000000)
OK

Running scenario 8/13 - Fixing mediator balance of the home mediator

Fixing mediator balance of the home mediator
Sending 10 tokens to the Home Mediator
pending tx: 0x2b2aa51c7bb017a1117340dec7178fed0ecaa31dd726a0dc7f85d9b7d131fde5
Balance diff 10000000000000000000
Sending fixMediatorBalance request to the Foreign Mediator
pending tx: 0xb21a8cf56104ae6b486a0d7520dde3b7291ee6cd72f2edc5ba73c543c2b060ce
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de7310000000000000666 to be processed
Getting address of the bridged token
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 10000000000000000000)
OK

Running scenario 9/13 - Fixing failed bridge operations on the home side

Getting address of the bridged token
Disabling execution for 0xDFa538D7D6aAB2Cb0fCF12657E070d3E6b42A06B
pending tx: 0x130c5528c161bb3a67204aaa108c911458e74d110dbcfcabca2d1eff4545da26
Sending 10 tokens to the Foreign Mediator
pending tx: 0x70565a307cc16b7484f5906dbca9e5551662a81901faad560896b2a8ebcfe20a
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b21326000000000000000ec9 to be processed
Requesting failed message fix for message id 0x000500009a6ff99b356dd998260582be7d95a4d08b21326000000000000000ec9
pending tx: 0xeab41d518c3d4cc0a54845e4f6765e9dbc7d6228918caa3e0eb300497f3970cf
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de7310000000000000667 to be processed
Checking if transaction has the required Transfer(0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 10000000000000000000)
Enabling back execution for 0xDFa538D7D6aAB2Cb0fCF12657E070d3E6b42A06B
```

```
pending tx: 0x96c9a28307526463efc90ed656251967563d4d7c97238e1c5881bcf8b8453cbb
Sending 10 tokens to the Home Mediator
pending tx: 0x7e8641a00cc806044025689ecce073c51287e9afd1135df02b2b6142e9220152
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de7310000000000000668 to be processed
Getting address of the bridged token
Disabling execution for 0x6acAB06915A93DD5dfBF10dA04D7175b5677D587
pending tx: 0x6f767a83565524f9c075106d98fa6f6d40d6c5d9f26faee8e2da762f563f2803
Sending 5 tokens to the Foreign Mediator
pending tx: 0xc5c018533b84ea1f79ae9c3517e85a3be141dae8ec9a4ff7398a55185712372a
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000eca to be processed
Requesting failed message fix for message id 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000eca
pending tx: 0x3f0b6e1040621575327777f47345b4dc3e01bbe04397089ccc18ce7043a53066
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de7310000000000000669 to be processed
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 5000000000000000000)
Enabling back execution for 0x6acAB06915A93DD5dfBF10dA04D7175b5677D587
pending tx: 0xc8a90f0d09ec15c59f7d2d176d67286e155fb53683b027f7de2931b138a3c1b5
OK

Running scenario 10/13 - Fixing failed bridge operations on the foreign side

Getting address of the bridged token
Disabling execution for 0xDFa538D7D6aAB2Cb0fCF12657E070d3E6b42A06B
pending tx: 0x0dda723d8fe861bf6642d4b80c33970a9e5cb711a062bad946c96dd4a72bc438
Sending 10 tokens to the Home Mediator
pending tx: 0x2bb2570e3b40f9dda8374c97f94c27f3cc6e11af1e6d4a5ecb898d7143a61d38
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de731000000000000066a to be processed
Requesting failed message fix for message id 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de731000000000000066a
pending tx: 0x13064e4f2004bc73bdb408d472263c18168928c5d89250ca1b7f89c2fe4d84c2
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000ecb to be processed
Checking if transaction has the required Transfer(0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 10000000000000000000)
```

```
Enabling back execution for 0xDFa538D7D6aAB2Cb0fCF12657E070d3E6b42A06B
pending tx: 0x45c14afcf22aa2667ef5d214606c16907f5c91c2081a6bbcd24e4a0325ebb87a
Sending 10 tokens to the Foreign Mediator
pending tx: 0x154cd0b67b09c1b121cc478d8f4cff72aa7925ce506dcfce1d3cc5110e8d58fd
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000ecc to be processed
Getting address of the bridged token
Disabling execution for 0x6acAB06915A93DD5dfBF10dA04D7175b5677D587
pending tx: 0x4f136aa992aa0dfa36981550bf4ff770638afe45f6bfb3a3c63e29ecf4081153
Sending 5 tokens to the Home Mediator
pending tx: 0x0c03d6ed81177da3187874554bd9d3fa767e966e36989d1facc23253e9bbb6ee
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de731000000000000066b to be processed
Requesting failed message fix for message id 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de731000000000000066b
pending tx: 0x484d141a988f481c143aecfcf3a1fe16268d14ce5409788dad93fa55e7dfd0bb
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000ecd to be processed
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 5000000000000000000)
Enabling back execution for 0x6acAB06915A93DD5dfBF10dA04D7175b5677D587
pending tx: 0x3a0f319509e8f78ed03687a088d31d877ce9bc3a2e9cdc3783d25af8661fe5de
OK
```

Running scenario 11/13 - Bridging of Foreign tokens with extra data

```
Bridging Native Foreign token to Home chain
Sending 10 tokens to the Foreign Mediator with extra data
pending tx: 0x2ba220f618756538145dea05970926dbde51334d5eb8b3c1c920a2df64d5ca3e
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000ece to be processed
Getting address of the bridged token
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x5Dcec429e1a46d8A8517DbA74E1733F763FB556F, 10000000000000000000)
Sending 5 tokens to the Home Mediator with extra data
pending tx: 0xe396fd17ef8160e87b5d18da769ba6300ba9407b9b0ba467884f50e89cf4fe6c
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de731000000000000066c to be processed
```

```
Checking if transaction has the required Transfer(0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,
0x5Dcec429e1a46d8A8517DbA74E1733F763FB556F, 5000000000000000000)
OK

Running scenario 12/13 - Bridging of Home tokens with extra data

Bridging Native Home token to Foreign chain
Sending 10 tokens to the Home Mediator with extra data
pending tx: 0xf24b9c52a98dd8c3019309f02e1f3996e4a6b606a8a42ae439292ec04d1e67c2
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de731000000000000066d to be processed
Getting address of the bridged token
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x5Dcec429e1a46d8A8517DbA74E1733F763FB556F, 10000000000000000000)
Sending 5 tokens to the Foreign Mediator with extra data
pending tx: 0x7122d44b8b079dfd2e6cd5f4b423729980a206efc6397711859a90174632be0f
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000ecf to be processed
Checking if transaction has the required Transfer(0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,
0x5Dcec429e1a46d8A8517DbA74E1733F763FB556F, 5000000000000000000)
OK

Running scenario 13/13 - Bridging of native Foreign ETH in both directions

Bridging Native Foreign ETH to Home chain
Sending 0.5 ETH to the Foreign Router
pending tx: 0x55fc19e140b2419524e35df3aec6ab327f6f4d0e426c1a2b847bc1a1e57443f1
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000ed0 to be processed
Getting address of the bridged token
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 500000000000000000)

Sending 0.5 more ETH to the Foreign Router
pending tx: 0xdcdf9d27897ce352e5df3684a4ab66f38440847afeacef345d66ca590fb23656
```

```
Waiting for message 0x000500009a6ff99b356dd998260582be7d95a4d08b2132600000000000000ed1 to be processed
Checking if transaction has the required Transfer(0x0000000000000000000000000000000000000000,
0x16016b0ACd9192eF51Af129aB75d110178c79C14, 500000000000000000)

Sending 0.5 bridged tokens to the Home Mediator
pending tx: 0x3b91f17b67dd1a0fd76445612d252523cfd83026794dbd9338cdaafe20900eb4
Waiting for message 0x00050000249bfc2f3cc8d68f6b6bf7230ea0a8ed853de731000000000000066e to be processed
Checking if transaction has the required Transfer(0xc4950F86E3c254Bda55eA52A2489399e2Fe230F5,
0x7Fa92e2419F6E5DC51aA2399b33b06d958De218E, 500000000000000000)
```
OK

Tests summary:
1) Claiming of foreign tokens - **OK**
2) Claiming of home tokens - **OK**
3) Bridging of native Foreign tokens in both directions - **OK**
4) Bridging of native Home tokens in both directions - **OK**
5) Bridging of native Foreign tokens in both directions with alternative receiver - **OK**
6) Bridging of native Home tokens in both directions with alternative receiver - **OK**
7) Fixing mediator balance of the foreign mediator - **OK**
8) Fixing mediator balance of the home mediator - **OK**
9) Fixing failed bridge operations on the home side - **OK**
10) Fixing failed bridge operations on the foreign side - **OK**
11) Bridging of Foreign tokens with extra data - **OK**
12) Bridging of Home tokens with extra data - **OK**
13) Bridging of native Foreign ETH in both directions – **OK**

Home Network: https://blockscout.com/poa/sokol/address/0x16016b0ACd9192eF51Af129aB75d110178c79C14/token-transfers
Foreign Network: https://kovan.etherscan.io/address/0x16016b0acd9192ef51af129ab75d110178c79c14#tokentxns

7. Verify claims

7.1 There is no such case when tokens were not actually locked/burnt but the message for AMB to deliver a bridge tokens request was sent.
Status: tested and verified ✅

7.2 No other ways to mint tokens which contracts are deployed by the OB Token Fabric. No other ways to unlock tokens transferred to the OB contract the methods listed in the claim #1. Only the AMB contract is authorised to call the OB contracts method listed above.
Status: tested and verified ✅

7.3 The OB contract executes the onTokenBridged method is safe manner so if the contract-recipient accidentally or purposely fails it does not affect the bridging operation: the tokens will be transferred to the recipient anyway. The OB logic is composed as so there is no way to build a request from the contract-recipient during the execution of onTokenBridged to force the OB contracts to mint/unlock extra tokens, execute another unauthorised action in the OB contract.
Status: tested and verified ✅

7.4 The pair of the methods requestFailedMessageFix and fixFailedMessage all ows to operate only with failed bridging operations. As part of the recovery operation, it is not possible to unlock/mint more tokens than was initially requested to be bridged. It is not possible to execute several times the recovery operation for the same failed bridge request. The AMB contract is only authorized to call the fixFailedMessage method.
Status: tested and verified ✅

7.5 These actions can be executed only by accounts authorised in advance.
Status: tested and verified ✅

7.6 Overall smart contract security and business logic needs to be checked
Status: tested and verified ✅

8. Executive Summary

Three (3) independent Chainsulting experts performed an unbiased and isolated audit of the OmniBridge codebase.
The main goal of the audit was to verify the claims regarding the security of the smart contract and the functions. During the audit, no critical issues were found, after the manual and automated security testing. Only informational and low issues were found, to increase the code quality. Overall, everything was well documented and worked as it was supposed to be.

9. Deployed Smart Contract

VERIFIED

Codebase:
https://github.com/poanetwork/omnibridge/releases/tag/1.0.0