

ABRACADABRA MONEY · SECURITY REVIEW

Abracadabra Money Arbitrum MIMv2 Migration.

A security review of Arbitrum MIMv2 Migration.

CLIENT

Abracadabra
Money

PROJECT

Arbitrum MIMv2
Migration

AUDITORS

Nicholas Chew,
Daniel Gelfand

AUDIT WINDOW

February 11 –
February 27, 2025

FINAL REPORT

February 27, 2025

§ 00

Contents.

42 pages: overview, confidence ranking, scope, 31 findings, disclaimer, and about Guardian.

§ 01	Overview	P. 03
§ 02	Confidence Ranking	P. 04
§ 03	Scope & Classifications	P. 05
§ 04	Findings	P. 06
	Main Review · 21 findings	P. 08
	Remediation Review · 10 findings	P. 29
§ 05	Disclaimer	P. 41
§ 06	About Guardian	P. 42

§ 01

Overview.

Guardian reviewed Arbitrum MIMv2 Migration across 2 review rounds. The final report includes 3 Medium, 28 Low findings.

AUDIT FIRM	METHODOLOGY
Guardian	Smart contract · Manual · Main Review · Remediation Review
CLIENT	
Abracadabra Money	
PROJECT	
Arbitrum MIMv2 Migration	
PREPARED BY	
Nicholas Chew, Daniel Gelfand	
AUDIT WINDOW	
February 11 – February 27, 2025	
FINAL REPORT	
February 27, 2025	

1.1 Confidence Ranking

5 / 5

Very High Confidence

Guardian assigns a Confidence Ranking of 5 out of 5, combining finding severity, remediation outcome, and the reviewer’s assessment of latent risk. The full scale is on the next page.

1	2	3	4	5
VERY LOW CONFIDENCE	LOW CONFIDENCE	MODERATE CONFIDENCE	HIGH CONFIDENCE	VERY HIGH CONFIDENCE

1.2 Findings at a Glance

SEVERITY	TOTAL	PENDING	ACK	PARTIAL	RESOLVED
■ Critical	0	0	0	0	0
■ High	0	0	0	0	0
■ Medium	3	0	1	0	2
■ Low	28	0	20	0	8
■ Informational	0	0	0	0	0
All severities	31	0	21	0	10

§ 02

Confidence Ranking.

Guardian assigns a project-level confidence score using the scale below. The ranking combines finding severity, remediation outcome, and the reviewer's assessment of latent risk.

CONFIDENCE RANKING	DEFINITION AND RECOMMENDATION	RISK PROFILE
5 Very High Confidence ASSIGNED	Codebase is mature, clean, and secure. No High or Critical vulnerabilities were found. Follows modern best practices with high test coverage and thoughtful design. Recommendation: code is highly secure at time of audit.	0 High/Critical findings and few Low/Medium severity findings.
4 High Confidence	Code is clean, well-structured, and adheres to best practices. Design patterns are sound and test coverage is strong. Recommendation: suitable for deployment after remediations; consider periodic review with changes.	0-1 High/Critical findings per engagement week and little to no Medium severity findings.
3 Moderate Confidence	Code is functional but shows gaps in defensive programming or test coverage. Recommendation: remediate the findings and re-review the changes before deployment.	Some High/Critical findings, with various Medium severity findings.
2 Low Confidence	Recurring issues suggest systemic gaps rather than isolated mistakes. Recommendation: remediate, then commission a follow-up review at a frozen commit.	Over 6 High and Critical findings across the engagement.
1 Very Low Confidence	The volume and severity of findings indicate the codebase is not ready for production. Recommendation: substantial rework, then a full re-review.	Over 12 High and Critical findings across the engagement.

§ 03

Scope & Classifications.

Guardian reviewed Arbitrum MIMv2 Migration.

IN SCOPE

contracts/AbraOFTUpgradeableExisting.sol

contracts/interfaces/IMintable.sol

deploy/MIMOFT.ts

3.1 Severity Matrix

	IMP · HIGH	IMP · MED	IMP · LOW
LIK · HIGH	Critical	High	Medium
LIK · MED	High	Medium	Low
LIK · LOW	Medium	Low	Low

IMPACT · HIGH

Significant loss of assets, harm to a group of users, or core protocol functionality is disrupted.

IMPACT · MEDIUM

Small amount of funds can be lost, or ancillary functionality is affected. Users may experience reduced or delayed receipt.

IMPACT · LOW

Unexpected behavior in some functionality that is notable but does not meet the criteria for higher severity.

LIKELIHOOD · HIGH

Possible with reasonable assumptions that mimic on-chain conditions; cost is low compared to amount gained.

LIKELIHOOD · MEDIUM

Possible only in uncommon cases or requires a large amount of capital relative to disruption caused.

LIKELIHOOD · LOW

Unlikely to ever occur in production.

§ 04

Findings.

Grouped by round. Visible IDs reset by severity within each round, so a finding is named the same way in the report as it was in the channel it was written in. Status is where each finding ended up; its own page says how.

ID	TITLE	CATEGORY	SEVERITY	STATUS	PAGE
MAIN REVIEW					
M-01	Excess Fees Are Unrefunded	Logical Error	MEDIUM	RESOLVED	p. 8
L-01	Minter Role Required	Warning	LOW	RESOLVED	p. 9
L-02	Incorrect Test Documentation	Documentation	LOW	RESOLVED	p. 10
L-03	Unbacked Dust Due to Shared Decimal Conversion	Warning	LOW	ACKNOWLEDGED	p. 11
L-04	Unexecuted User Sends Will Not Receive their V1 Funds	Warning	LOW	ACKNOWLEDGED	p. 12
L-05	Total Supply Can Be Modified Before Migration	Warning	LOW	ACKNOWLEDGED	p. 13
L-06	Incorrect Addresses	Warning	LOW	RESOLVED	p. 14
L-07	Mint-Bridge-Disable Must Be Done Atomically	Warning	LOW	ACKNOWLEDGED	p. 15
L-08	Other Alt-Chains Not Wired	Warning	LOW	ACKNOWLEDGED	p. 16
L-09	Migration With Zero Supply	Optimization	LOW	ACKNOWLEDGED	p. 17
L-10	Transfers To MIM Contract Block Messages	Validation	LOW	ACKNOWLEDGED	p. 18
L-11	Zero Address Transfers	Documentation	LOW	ACKNOWLEDGED	p. 19
L-12	Extraneous Operators Must Be Removed	Warning	LOW	ACKNOWLEDGED	p. 20
L-13	Minter Role Required	Warning	LOW	RESOLVED	p. 21
L-14	Msg.value Is Lost With lzReceive	Warning	LOW	ACKNOWLEDGED	p. 22
L-15	Mint-Bridge-Disable Must Be Done Atomically	Warning	LOW	ACKNOWLEDGED	p. 23
L-16	Step 3 Should Be Completed After Successful Transfer	Documentation	LOW	RESOLVED	p. 24
L-17	Re-org Risk	Warning	LOW	ACKNOWLEDGED	p. 25
L-18	Incompatibility With Existing Protocols	Documentation	LOW	ACKNOWLEDGED	p. 26
L-19	Extraneous Operators Must Be Removed	Warning	LOW	ACKNOWLEDGED	p. 27
L-20	Zero Address Transfers	Documentation	LOW	ACKNOWLEDGED	p. 28
REMEDIATION REVIEW					
M-01	FeeHandler Calls Bricked	DoS	MEDIUM	RESOLVED	p. 29

Findings continued.

ID	TITLE	CATEGORY	SEVERITY	STATUS	PAGE
M-02	totalSupply Does Not Reflect Pending Failed LZ Messages	Warning	MEDIUM	ACKNOWLEDGED	p. 30
L-01	Token May Be Misleading	Documentation	LOW	RESOLVED	p. 32
L-02	Existing OFTs Will Not Refund Excess Native	Documentation	LOW	ACKNOWLEDGED	p. 33
L-03	Optimism FeeHandler Will Revert With New OFT	Warning	LOW	RESOLVED	p. 34
L-04	Address Verification	Documentation	LOW	RESOLVED	p. 35
L-05	TODO In Optimism Config	Warning	LOW	ACKNOWLEDGED	p. 36
L-06	Overly Conservative Confirmation Setting for Berachain	Warning	LOW	ACKNOWLEDGED	p. 37
L-07	Missing Validation in setFeeTo Function	Warning	LOW	ACKNOWLEDGED	p. 39
L-08	Incomplete Cross-Chain Configuration for Alt-Chains	Warning	LOW	ACKNOWLEDGED	p. 40

MEDIUM **M-01**

RESOLVED

Excess Fees Are Unrefunded

CATEGORY	Logical Error
LOCATION	AbraOFTUpgradeableExisting.sol
REVIEW	Main Review

DESCRIPTION

The AbraOFTUpgradeableExisting contract inherits from SenderWithFees, which overrides the default behavior of `_lzSend`. Unlike the standard implementation where `msg.value` must match the quoted `fee.nativeFee`, the override allows `msg.value` to be any amount.

This introduces several issues:

- Excess funds are not refunded – If a user sends more than required, the excess remains stuck in the AbraOFTUpgradeableExisting contract.
- Users can bypass fee payments – If the contract has a balance of native tokens, users can send transactions with `msg.value == 0`, causing the contract to cover their fees.

RECOMMENDATION

Modify the overridden `_lzSend` to validate that `msg.value == nativeFee + protocolNativeFee`.

RESOLUTION

Status: Resolved.

Guardian: The `quoteNativeFee` function accepts the `_fee.nativeFee` and `_fee.lzTokenFee` parameters from the user-supplied `MessagingFee`. However, the `msgValue` is disjoint from the `MessagingFee` since there is no validation that they must match as in LayerZero's default `_lzSend`.

Depending on the `FeeHandler` implementation, it would be possible for a user to pass a zero'd `MessagingFee` so that they are not charged `protocolNativeFees`, yet still pass a normal amount of `msg.value` for the OFT transfer to succeed, ultimately bypassing the protocol fee.

****Recommendation****

Ensure that `FeeHandlers` do not rely on the user-supplied `MessagingFee`.

<https://github.com/Abacadabra-money/abracadabra-oft-v2/pull/6/commits/c3ac8a600e9ad29530d71584ce6d4b03a65dd4c5>

LOW **L-01**

RESOLVED

Minter Role Required

CATEGORY	Warning
LOCATION	AbraOFTUpgradeableExisting.sol
REVIEW	Main Review

DESCRIPTION

For mint and burn functionality to be permissible on the AnyswapV5 MIM token, which is necessary for debit/credit to facilitate cross-chain transfers, the AbraOFTUpgradeableExisting OFT needs to be set as a minter such that `isMinter = true`. However, calling `setMinter` and `applyMinter` from the [Vault] (<https://arbiscan.io/address/0xf46BB6dDA9709C49EfB918201D97F6474EA5Aea>) contract would not work on Kava, Base, Linea, Blast since MintableBurnableERC20 token is used instead of AnyswapV5ERC20.

Furthermore, the current migration plan is to set the AbraOFTUpgradeableExisting OFT as an operator on the ElevatedMintedBurner, which would not allow bridging since the AbraOFTUpgradeableExisting contract interacts with the MIM token directly e.g. `token_.mint(_to, _amountLD);`

RECOMMENDATION

Modifiy AbraOFTUpgradeableExisting's `_debit` and `_credit` functions such that the ElevatedMintedBurner's `mint` and `burn` functions are used instead of interacting with the token directly.

RESOLUTION

Status: Resolved.

LOW **L-02**

RESOLVED

Incorrect Test Documentation

CATEGORY Documentation
LOCATION Abra0FTForkTests.t.sol: 93
REVIEW Main Review

DESCRIPTION

The listed test commands such as `forge test --fork-url [ https://arb1.arbitrum.io/rpc ]` (`https://arb1.arbitrum.io/rpc`) `--mc AbraForkTests` are no longer applicable since the contract has been renamed to `AbraForkTestBase`.

RECOMMENDATION

Adjust the testing documentation.

RESOLUTION

Status: Resolved.

<https://github.com/Abracadabra-money/abracadabra-oft-v2/pull/6/commits/f97def916d7677170936f012ac9a97a37bab8464>

LOW **L-03**

ACKNOWLEDGED

Unbacked Dust Due to Shared Decimal Conversion

CATEGORY Warning
LOCATION Migration Doc
REVIEW Main Review

— DESCRIPTION

When the newly minted Arbitrum OFT `totalSupply` is bridged to mainnet, some dust remains due to the conversion from local to shared decimals, division by $1e10$ in this case. Consequently, a small amount of dust will not be bridged over and will be unbacked since slightly less is received on mainnet. Furthermore, the `totalSupply` of the Arbitrum OFT afterwards will be slightly greater than it was prior to the "Mint Total Arbitrum Supply" migration step.

— RECOMMENDATION

Clearly document this behavior or account for the dust with a privileged mint.

— RESOLUTION

Status: Acknowledged.

LOW **L-04**

ACKNOWLEDGED

Unexecuted User Sends Will Not Receive their V1 Funds

CATEGORY	Warning
LOCATION	Migration Doc
REVIEW	Main Review

— DESCRIPTION

User who have sent their funds before the bridges were closed are at risk of not receiving their funds. Consider the following example:

- (1) Bob sends 50 MIM to Arbitrum from Mainnet, the packet is sent but unexecuted yet.
- (2) Abracadabra begins migration and closes the bridges by disabling minting/burning.
- (3) Bob's message to transfer 50 MIM cannot be executed as `lzReceive` would revert.
- (4) Bob ultimately loses his funds as the LayerZeroV1 OFT's are no longer operational and the funds will be stuck in the LayerZeroV1 Proxy on mainnet.

Note that this also applies to cross-chain transfers between the other bridges.

— RECOMMENDATION

Clearly document this risk to users during the migration process. Furthermore, ensure users are informed that mainnet will allow `lzSend` even if the altchain OFT's are disabled.

— RESOLUTION

Status: Acknowledged.

LOW **L-05**

ACKNOWLEDGED

Total Supply Can Be Modified Before Migration

CATEGORY Warning
LOCATION Migration Doc
REVIEW Main Review

— DESCRIPTION

Users are able to burn their own `AnyswapV5ERC20` ** **tokens when `_vault0nly` is false by using function `Swapout` .

If the migration precalculates the supply that needs to be minted, a user can frontrun to burn some MIM. Consequently, the wrong amount of MIM would be minted and a greater amount of MIM would be extracted from the LZV1 OFT than exists on Arbitrum. However, this is not a large issue since the LZV2 adapter will still hold enough backing tokens, and the excess amount that was sent is effectively burnt.

— RECOMMENDATION

Consider setting `_vault0nly` to true or clearly document this behavior.

— RESOLUTION

Status: Acknowledged.

LOW **L-06**

RESOLVED

Incorrect Addresses

CATEGORY	Warning
LOCATION	Migration Doc
REVIEW	Main Review

DESCRIPTION

The current listed KAVA OFT address is `0x6e858b0DD9a9Dcdf710B28C236292E30ba079728` but that address has no contract on Kava. In fact, that is the address of the Moonriver ElevatedMinterBurner. The proper address should be `0xc7a161cfd0e133d289b13692b636b8e8b5cd8d8c`

The current listed Arbitrum MinterBurner address is `0x957A8Af7894E76e16DB17c2A913496a4E60B7090` but that is really the OFT's address. Instead it should be address

`0x26F20d6Dee51ad59AF339BEf9f721113D01b6b3`

The current listed BSC MinterBurner address is `0x41D5A04B4e03dC27dC1f5C5A576Ad2187bc601Af` but that is really the OFT's address. Instead it should be address `0x79533F85479e04d2214305638B6586b724beC951`

RECOMMENDATION

Ensure to update the migration document with the proper addresses.

RESOLUTION

Status: Resolved.

LOW **L-07**

ACKNOWLEDGED

Mint-Bridge-Disable Must Be Done Atomically

CATEGORY	Warning
LOCATION	Migration Doc
REVIEW	Main Review

— DESCRIPTION

Currently Step 2 of the migration is not specified to be atomic. If this step is not atomic, the following scenario could arise:

- (1) 150 AnyswapV5 MIM tokens are sitting on Arbitrum, 50 AnyswapV5 MIM tokens are sitting on Fantom, 200 MIM within ETH LzProxyOFTV2
- (2) Once minting is allowed, 150 MIM are minted and bridged to ETH. ETH LzProxyOFTV2 now only holds 50 MIM and new adapter contract holds the 150 MIM.
- (3) 50 MIM from FTM is received by the Arbitrum OFT, for example a prior failed message is retried. Now Arbitrum has 200 MIM, but 50 of those are backed by the LayerZero V1 contracts.
- (4) Because the LayerZero V1 OFT connections are then disabled, those 50 MIM sitting within ETH LzProxyOFTV2 cannot be retrieved, and the Arbitrum MIM does not have proper backing.

— RECOMMENDATION

Perform this step atomically for all chains and document this in the migration plan.

— RESOLUTION

Status: Acknowledged.

LOW **L-08**

ACKNOWLEDGED

Other Alt-Chains Not Wired

CATEGORY Warning
LOCATION Migration Doc
REVIEW Main Review

— DESCRIPTION

The migration is for OFT's across multiple altchains, however connections are only configured for the new Arbitrum OFT. This may be unexpected for users who desire to have cross-chain functionality from their new OFT's on chains such as Fantom, Polygon, etc.

— RECOMMENDATION

Ensure pathways are configured between all new LayerZero V2 OFT's.

— RESOLUTION

Status: Acknowledged.

LOW **L-09**

ACKNOWLEDGED

Migration With Zero Supply

CATEGORY Optimization
LOCATION Migration Doc
REVIEW Main Review

— DESCRIPTION

Currently the migration steps would mint and bridge even if a chain has zero MIM supply. Although this is an extremely rare case, gas can be saved on not attempting these actions.

— RECOMMENDATION

Consider skipping mint and bridge if there is zero supply or document this.

— RESOLUTION

Status: Acknowledged.

LOW **L-10**

ACKNOWLEDGED

Transfers To MIM Contract Block Messages

CATEGORY	Validation
LOCATION	OFTCoreUpgradeable.sol: 212, 282
REVIEW	Main Review

— DESCRIPTION

The AnyswapV5ERC20 MIM contract does not allow transfers to itself (the MIM contract). However, in the AbraOFTUpgradeableExisting contract, there is no validation preventing users from initiating cross-chain composed messages which would send tokens to the destination chain MIM token contract. As a result any initiated transfers with the to address as the MIM token on the destination chain will be un-executable.

— RECOMMENDATION

This may be resolved by storing the address of the MIM token contract on each corresponding dstEid in a mapping and validating that the to address is not the destination's MIM contract by overriding the send function.

Alternatively, the `_lzReceive` function could include logic to send the bridged tokens to a trusted holding address if the to address is the MIM token contract.

If no code changes should be implemented for this issue, be sure to clearly document and warn users of this risk.

— RESOLUTION

Status: Acknowledged.

LOW **L-11**

ACKNOWLEDGED

Zero Address Transfers

CATEGORY	Documentation
LOCATION	Migration Doc
REVIEW	Main Review

— DESCRIPTION

The LzProxyOFTV2 on Mainnet performs a plain transfer to the recipient. Before bridging from altchains is disabled, a user can send a transfer to mainnet with the `address(0)` recipient. This message will never execute, and consequently these funds will forever be locked in the LzProxyOFTV2 contract. Note that this isn't an attack vector, since these funds are effectively "burnt" by the user from the altchain and hence should not be represented in the backing of the new LayerZeroV2 OFT. Also note that the latest Adapter OFT on mainnet also does not redirect zero address transfers.

— RECOMMENDATION

Be aware of this behavior.

— RESOLUTION

Status: Acknowledged.

LOW **L-12**

ACKNOWLEDGED

Extraneous Operators Must Be Removed

CATEGORY	Warning
LOCATION	Migration Doc
REVIEW	Main Review

— DESCRIPTION

During the MIM V2 migration, it is critical to lock the total supply of MIM. Throughout the altchains, MIM may have more than a single operator. For example, on altchains that use the Anyswap token, in contrast to Kava, Linea, Base and Blast which use MintableBurnableERC20, the Anyswap router should be removed as an operator during the migration process. Furthermore, the Moonriver MIM contract had duplicate minter rights granted to Multichain, therefore steps should be taken to revoke all of them.

— RECOMMENDATION

For all chains remove any unnecessary operators to prevent supply changes during the migration process. Ensure only trusted contracts retain minting control during the migration process.

— RESOLUTION

Status: Acknowledged.

LOW **L-13**

RESOLVED

Minter Role Required

CATEGORY	Warning
LOCATION	AbraOFTUpgradeableExisting.sol
REVIEW	Main Review

DESCRIPTION

For mint and burn functionality to be permissible on the AnyswapV5 MIM token, which is necessary for debit/credit to facilitate cross-chain transfers, the AbraOFTUpgradeableExisting OFT needs to be set as a minter such that `isMinter = true`. However, calling `setMinter` and `applyMinter` from the [Vault] (<https://arbiscan.io/address/0xf46BB6dDA9709C49EfB918201D97F6474EAc5Aea>) contract would not work on Kava, Base, Linea, Blast since MintableBurnableERC20 token is used instead of AnyswapV5ERC20.

Furthermore, the current migration plan is to set the AbraOFTUpgradeableExisting OFT as an operator on the ElevatedMintedBurner, which would not allow bridging since the AbraOFTUpgradeableExisting contract interacts with the MIM token directly e.g. `token_.mint(_to, _amountLD);`

RECOMMENDATION

Modify AbraOFTUpgradeableExisting's `_debit` and `_credit` functions such that the ElevatedMintedBurner's `mint` and `burn` functions are used instead of interacting with the token directly.

RESOLUTION

Status: Resolved.

LOW **L-14**

ACKNOWLEDGED

Msg.value Is Lost With lzReceive

CATEGORY	Warning
LOCATION	AbraOFTUpgradeableExisting.sol, AbraOFTAdapterUpgradeable.sol
REVIEW	Main Review

— DESCRIPTION

If users specify `msg.value` in the message execution options, this value will be passed by the executor while calling `lzReceive` on the destination chain. The value will be lost as it just ends up as a balance of `AbraOFTUpgradeableExisting` or `AbraOFTAdapterUpgradeable` with no way of retrieving it.

— RECOMMENDATION

The balance of `AbraOFTUpgradeableExisting` or `AbraOFTAdapterUpgradeable` can be retrieved by adding an admin function to transfer the balance to the owner. Otherwise, make sure to document this behavior in the docs.

— RESOLUTION

Status: Acknowledged.

LOW **L-15**

ACKNOWLEDGED

Mint-Bridge-Disable Must Be Done Atomically

CATEGORY	Warning
LOCATION	Migration Doc
REVIEW	Main Review

— DESCRIPTION

Currently Step 2 of the migration is not specified to be atomic. If this step is not atomic, the following scenario could arise:

- (1) 150 AnyswapV5 MIM tokens are sitting on Arbitrum, 50 AnyswapV5 MIM tokens are sitting on Fantom, 200 MIM within ETH LzProxyOFTV2
- (2) Once minting is allowed, 150 MIM are minted and bridged to ETH. ETH LzProxyOFTV2 now only holds 50 MIM and new adapter contract holds the 150 MIM.
- (3) 50 MIM from FTM is received by the Arbitrum OFT, for example a prior failed message is retried. Now Arbitrum has 200 MIM, but 50 of those are backed by the LayerZero V1 contracts.
- (4) Because the LayerZero V1 OFT connections are then disabled, those 50 MIM sitting within ETH LzProxyOFTV2 cannot be retrieved, and the Arbitrum MIM does not have proper backing.

— RECOMMENDATION

Perform this step atomically for all chains and document this in the migration plan.

— RESOLUTION

Status: Acknowledged.

LOW **L-16**

RESOLVED

Step 3 Should Be Completed After Successful Transfer

CATEGORY Documentation

LOCATION Migration Doc

REVIEW Main Review

— DESCRIPTION

During the migration, step 3 to activate the MIM bridges should be performed after the messages from step 2 are successfully received and the adapter properly backs the MIMV2 OFT on the altchains. This is necessary because if a user transfers their MIM from an altchain to Mainnet before step 2 messages are fulfilled, the message receipt will either revert or consume the backing of other chains such as Bera.

— RECOMMENDATION

Document this in the migration plan.

— RESOLUTION

Status: Resolved.

LOW **L-17**

ACKNOWLEDGED

Re-org Risk

CATEGORY	Warning
LOCATION	Migration Doc
REVIEW	Main Review

— DESCRIPTION

Certain chains are susceptible to block re-orgs such as Polygon. In the case of a block re-org, the transaction for step 2 may end up being executed before the transaction for step 1. This may lead the total supply to be minted and bridged when pre-crime is still enabled, which is expected to be turned off according to the migration plan. Furthermore, if step 2 executes before step 1, a user may seize the opportunity to execute their pending message payload since the OFT would not be disabled as an operator yet. This could lead to too little MIM being bridged over to Mainnet when step 2 executes with a stale total supply, leaving a portion of funds unbacked from the mainnet v2 adapter.

— RECOMMENDATION

Be aware of this case or consider waiting for sufficient confirmations before proceeding with the next step.

— RESOLUTION

Status: Acknowledged.

LOW **L-18**

ACKNOWLEDGED

Incompatibility With Existing Protocols

CATEGORY Documentation

LOCATION Migration Doc

REVIEW Main Review

— DESCRIPTION

LayerZeroV1 OFT's rely on function `sendFrom` for crosschain transfers, while LayerZeroV2 OFT's use function `send`. Consequently, any existing protocols that rely on function `sendFrom` will experience reverts.

— RECOMMENDATION

Ensure protocols are aware of the migration, that their operations may be affected, and will need to update their calls to use `send`

— RESOLUTION

Status: Acknowledged.

LOW **L-19**

ACKNOWLEDGED

Extraneous Operators Must Be Removed

CATEGORY	Warning
LOCATION	Migration Doc
REVIEW	Main Review

— DESCRIPTION

Throughout the altchains, MIM may have more than a single operator. For example, on altchains that use the Anyswap token, in contrast to Kava, Linea, Base and Blast which use MintableBurnableERC20, the Anyswap router should be removed as an operator during the migration process.

— RECOMMENDATION

For all chains remove any unnecessary operators to prevent supply changes during the migration process.

— RESOLUTION

Status: Acknowledged.

LOW **L-20**

ACKNOWLEDGED

Zero Address Transfers

CATEGORY	Documentation
LOCATION	Migration Doc
REVIEW	Main Review

— DESCRIPTION

The LzProxyOFTV2 on Mainnet performs a plain transfer to the recipient. Before bridging from altchains is disabled, a user can send a transfer to mainnet with the `address(0)` recipient. This message will never execute, and consequently these funds will forever be locked in the LzProxyOFTV2 contract. Note that this isn't an attack vector, since these funds are effectively "burnt" by the user from the altchain and hence should not be represented in the backing of the new LayerZeroV2 OFT. Also note that the latest Adapter OFT on mainnet also does not redirect zero address transfers.

— RECOMMENDATION

Be aware of this behavior and document this for users.

— RESOLUTION

Status: Acknowledged.

MEDIUM **M-01**

RESOLVED

FeeHandler Calls Bricked

CATEGORY DoS
LOCATION SenderWithFees.sol
REVIEW Remediation Review

DESCRIPTION

AbraOFTUpgradeableExisting.sol is currently set to be deployed on Arbitrum with `0xE66BE95FE4E3889a66925d996AF3E4dC173754a2` as the FeeHandler. The issue is that this FeeHandler does not support the new `quoteNativeFee()` interface without parameters, which will lead to a revert when sending any cross-chain transfer.

RECOMMENDATION

Deploy the updated FeeHandler and then update the `deploy/MIMOFT.ts` script to use the new address.

RESOLUTION

Status: Resolved.

MEDIUM **M-02**

ACKNOWLEDGED

totalSupply Does Not Reflect Pending Failed LZ Messages

CATEGORY	Warning
LOCATION	Global
REVIEW	Remediation Review

— DESCRIPTION

During the migration to the MIM v2 token standard, the `setOperator(<OFT>, false)` function is called on all alternate chains (alt-chains) to revoke operator permissions. However, due to the asynchronous nature of blockchain networks, these calls cannot be executed simultaneously across all chains. This creates a timing window where cross-chain bridging fails, leading to inaccurate token supply tracking.

As part of the migration process, the `setOperator(<OFT>, false)` function is invoked on each alt-chain to disable operator permissions for the Omnichain Fungible Token (OFT) contract. For example:

- On Arbitrum, this call occurs at timestamp 1000.
- On Optimism (and potentially other alt-chains), it's delayed, occurring at timestamp 1020 due to network differences.

If a user bridges MIM tokens from Optimism to Arbitrum at timestamp 1007, after Arbitrum's operator is revoked but before Optimism's, the transaction fails:

- On Optimism, the tokens are locked or burned to initiate the transfer.
- On Arbitrum, the `lzReceive` function reverts because the operator permissions were revoked at timestamp 1000, blocking the minting of tokens.

This results in:

1. The transfer doesn't complete.
2. Arbitrum's `totalSupply` doesn't reflect the pending tokens, as they aren't minted.
3. The MIM v2 Mainnet Adapter (`AbraOFTAdapterUpgradeable`) fails to fully back alt-chains if these pending tokens remain unprocessed.

Pending messages can be retried later, but until they are, the supply on Arbitrum (and potentially other alt-chains) remains understated.

— RECOMMENDATION

When the new MIM amount to send to the MIM v2 Mainnet Adapter is calculated, do consider the MIM that is part of these failed/pending messages to guarantee that the alt chain supply is fully backed.

M-02 totalSupply Does Not Reflect Pending Failed LZ Messages

CONTINUED

— RESOLUTION

Status: Acknowledged.

LOW **L-01**

RESOLVED

Token May Be Misleading

CATEGORY Documentation
LOCATION Abra0FTUpgradeableExisting.sol
REVIEW Remediation Review

DESCRIPTION

With the `token_` of `Abra0FTUpgradeableExisting` being the ElevatedMinterBurner on chains using the Anyswap MIM token, the `token()` function may return misleading results to integrators.

RECOMMENDATION

Clearly document this behavior.

RESOLUTION

Status: Resolved.

<https://github.com/Abacadabra-money/abracadabra-oft-v2/pull/6/commits/088629baf601c6175cd8ce9835666c042e795c29>

LOW **L-02**

ACKNOWLEDGED

Existing OFTs Will Not Refund Excess Native

CATEGORY	Documentation
LOCATION	Global
REVIEW	Remediation Review

— DESCRIPTION

The live version of the Mainnet LZ adapter as well as the Berachain OFT use `SenderWithFees` where `msg.value` in excess of the `MessagingFee` is not refunded and will remain in the contract. This may lead to loss of funds for unsuspecting users who expect a typical refund or validation that they did not send too much or too little native tokens for a cross-chain transfer.

— RECOMMENDATION

Clearly document this behavior for the existing LayerZero V2 OFT's.

— RESOLUTION

Status: Acknowledged.

LOW **L-03**

RESOLVED

Optimism FeeHandler Will Revert With New OFT

CATEGORY	Warning
LOCATION	Global
REVIEW	Remediation Review

DESCRIPTION

The FeeHandler on Optimism `0x0B79A169bF618bDD2a73698Fc8ec4A451182a012` (deployed on February 20, 2025) does not have a `quoteNativeFee` function without parameters. Consequently, `[FeeHandler Calls Bricked]`(<https://app.notion.com/p/1a28bda5828c805c97c0e4b7f160c46e>) will apply on this chain if it was set as the FeeHandler.**

RECOMMENDATION

Deploy a new FeeHandler with the updated `quoteNativeFee` function and only then set it for the OFT.

RESOLUTION

Status: Resolved.

LOW **L-04**

RESOLVED

Address Verification

CATEGORY	Documentation
LOCATION	Global
REVIEW	Remediation Review

DESCRIPTION

The `0x29d2fEC890B037B2d34f061F9a50f76F85ddBcAE` BERA/USD feed on Berachain is unverified. Furthermore, it has zero activity.

Furthermore, safe address on Optimism is `0xCbb86ffF0F8094C370cdDb76C7F270C832a8C7C0` but the address has limited activity.

RECOMMENDATION

Ensure the above addresses are Abra controlled and valid.

RESOLUTION

Status: Resolved.

LOW **L-05**

ACKNOWLEDGED

TODO In Optimism Config

CATEGORY Warning
LOCATION `hardhat.config.ts`
REVIEW Remediation Review

— DESCRIPTION

The `optimism-mainnet` config currently has a TODO as the `safeUrl` which may impact LayerZero wiring operations.

— RECOMMENDATION

Be aware of this and ensure it is updated.

— RESOLUTION

Status: Acknowledged.

LOW **L-06**

ACKNOWLEDGED

Overly Conservative Confirmation Setting for Berachain

CATEGORY	Warning
LOCATION	bracadabra-oft-v2-existing-token-oft/layerzero.mim.config.ts
REVIEW	Remediation Review

DESCRIPTION

Berachain employs a "single slot finality" mechanism, as outlined in its [documentation] (<https://docs.berachain.com/learn/help/faqs>) and [glossary] (<https://docs.berachain.com/learn/help/glossary#finality>). This means that once a block is produced in a slot, it is immediately considered final and transactions within it are irrevocably settled without requiring additional confirmations. This instant finality ensures that confirmed transactions cannot be altered or reversed, providing a high level of assurance right away.

However, in the LayerZero configuration script for Abracadabra's MIM token (`bracadabra-oft-v2-existing-token-oft/layerzero.mim.config.ts`), the confirmation requirement for messages sent from Berachain to other chains, such as Ethereum and Arbitrum, is set to 20 blocks. This is evident in the configuration:

```
``typescript
const connections = await generateConnectionsConfig([
  [ethereumContract, beraContract, [['LayerZero Labs', 'MIM'], []], [15, 20],
  [EVM_ENFORCED_OPTIONS, EVM_ENFORCED_OPTIONS]],
  [ethereumContract, arbitrumContract, [['LayerZero Labs', 'MIM'], []], [15, 20],
  [EVM_ENFORCED_OPTIONS, EVM_ENFORCED_OPTIONS]],
  [arbitrumContract, beraContract, [['LayerZero Labs', 'MIM'], []], [20, 20],
  [EVM_ENFORCED_OPTIONS, EVM_ENFORCED_OPTIONS]],
]) as OmniEdgeHardhat<0AppEdgeConfig>[];
```

Here, the `[15, 20]` for `[ethereumContract, beraContract]` indicates 15 confirmations for messages from Ethereum to Berachain and 20 confirmations for messages from Berachain to Ethereum. Similarly, `[20, 20]` for `[arbitrumContract, beraContract]` sets 20 confirmations for messages from Berachain to Arbitrum (and vice versa). This setting of 20 confirmations for messages originating from Berachain appears overly conservative given its instant finality.

L-06 Overly Conservative Confirmation Setting for Berachain

CONTINUED**— RECOMMENDATION**

Merely informative. If higher bridging speed is required, be aware that the confirmation requirement for messages sent from Berachain to other chains can be reduced, leveraging Berachain's single slot finality.

— RESOLUTION

Status: Acknowledged.

LOW **L-07**

ACKNOWLEDGED

Missing Validation in setFeeTo Function

CATEGORY	Warning
LOCATION	FeeHandler.sol: 86
REVIEW	Remediation Review

DESCRIPTION

The `setFeeTo` function in the contract is designed to set an address (`feeTo`) that receives protocol fees in the form of native assets (e.g., Ether) during cross-chain operations. However, it lacks proper validation to ensure the specified `feeTo` address can actually receive these native assets. As the protocol relies on a low-level call in the `_lzSend` function to transfer fees, if the `feeTo` address can not receive native assets (i.e. if it doesn't implement a `receive` function) the bridging would be totally blocked:

```
``solidity
(bool success, ) = IFeeHandler(feeHandler).feeTo().call{ value: protocolNativeFees }("");
``
```

RECOMMENDATION

Consider adding a check in `setFeeTo` function that performs a zero-value transfer to the proposed `feeTo` address. This ensures the address can receive native assets without requiring the contract to hold funds.

RESOLUTION

Status: Acknowledged.

LOW **L-08**

ACKNOWLEDGED

Incomplete Cross-Chain Configuration for Alt-Chains

CATEGORY Warning
LOCATION `layerzero.mim.config.ts`
REVIEW Remediation Review

— DESCRIPTION

The configuration file `layerzero.mim.config.ts` lacks a connection definition for Optimism, and potentially other alt-chains, despite evidence in the deployment script (`MIM0FT.ts`) that these chains are intended to be supported. Without these connections, cross-chain bridging between Mainnet and Optimism (or other alt-chains) cannot occur, as LayerZero relies on explicitly defined pathways to facilitate messaging and token transfers.

— RECOMMENDATION

Update `layerzero.mim.config.ts` to include Optimism and other intended alt-chains in the `generateConnectionsConfig` array. Use appropriate confirmation numbers and options based on chain characteristics (e.g., Optimism's finality).

— RESOLUTION

Status: Acknowledged.

§ 05

Disclaimer.

This report is not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team or project that contracts Guardian to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance.

This report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. Guardian's position is that each company and individual are responsible for their own due diligence and continuous security. Guardian's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.

The assessment services provided by Guardian is subject to dependencies and under continuing development. You agree that your access and/or use, including but not limited to any services, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis.

Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives, false negatives, and other unpredictable results. The services may access, and depend upon, multiple layers of third-parties.

Notice that smart contracts deployed on the blockchain are not resistant from internal/external exploit. Notice that active smart contract owner privileges constitute an elevated impact to any smart contract's safety and security. Therefore, Guardian does not guarantee the explicit security of the audited smart contract, regardless of the verdict.

§ 06

About Guardian.

Guardian is a smart contract security firm. Every engagement is worked by two independent teams competing on the same scope, each with its own tooling and methods, so no single reviewer's blind spot survives.

\$45B+DIGITAL ASSETS UNDER
REVIEW**200+**

ENGAGEMENTS COMPLETED

500+CRITICAL & HIGH FINDINGS
REPORTED

Reach us at audits@guardianaudits.com.