
Etherlink - LayerZero V2 - WXTZ

Tezos Foundation

Independent security assessment
report

inference



Report version: 1.0 / date: 22.07.2025

Table of contents

Table of contents	2
Summary	4
WXTZ	4
Overview on findings	5
Project overview	6
Scope	6
Scope limitations	6
Objectives	7
Activities	7
Findings	8
ELO-001: Message library not explicitly set	9
ELO-002: Not all used message libraries explicitly configured	10
ELO-003: Impossibility to identify an audited commit	11
ELO-004: No multisig configured for WXTZ owner role	12
ELO-005: DVN configuration and Etherlink commitment refutation	13
ELO-006: Known security bugs in used compiler version	14
Disclaimer	15
Appendix	17
Initial assessment	17
WXTZ' LayerZero integration	17
WXTZ configuration	28
DVNs	31
Multisig setup	33
Deployed bytecode verification	36
Solidity compiler version	41
Bridge solution verification	41
Reassessment	45
Message library configuration	45
Multisig setup	45
DVN configuration and Etherlink commitment refutation	49
Owner / delegate permission	51
Functions	51

Risks	51
Risk rating definition	53
Glossary	54

Version / Date	Description
0.1 / 28.11.2024	Initial draft version of the report
0.2 / 17.01.2025	<ul style="list-style-type: none"> Updated with results of reassessment of ELO-001 and ELO-002.
0.3 / 27.01.2025	<ul style="list-style-type: none"> Updated with the result of the transfer test of WXTZ across supported blockchains.
0.4 / 21.02.2025	<ul style="list-style-type: none"> Updated with the result of the reassessment after the ownership transfer to a multisig solution. Updated with the result of the reassessment of the DVN configuration and Etherlink commitment refutation
0.5 / 24.02.2025	<ul style="list-style-type: none"> Adding information about the reassessment of ELO-001 and ELO-002 in the appendix.
0.6 / 15.07.2025	<ul style="list-style-type: none"> Updated with the result of the reassessment of ELO-003.
1.0 / 22.07.2025	Final version



Summary

Inference AG was engaged by the Tezos Foundation (hereinafter “TF”) to perform an independent security assessment of the WXTZ setup for Etherlink, which is built on the LayerZero V2 technology stack.

Inference AG performed the security assessment based on the agreed scope, following our approach and activities as outlined in the “[Project overview](#)” chapter between 09/11/2024 and 22/07/2025.

Based on our activities, we identified several findings, ranging from high severity to informational. We recommend carefully analyzing these findings and defining appropriate measures to address the associated risks.

WXTZ

While the core code of LayerZero V2 (specifically the endpoint and message libraries) is immutable, the operation of the WXTZ bridge solution also includes security-critical roles held by the configured DVN operators and the WXTZ bridge owner role.

In section [Risks](#) in the Appendix, we outline a list of potential high-risk scenarios that the WXTZ bridge owner role could execute. As a result, users of the WXTZ bridge and holders of WXTZ must place significant trust in the bridge owner role, which is currently secured only by an Externally Owned Account (EOA).

The WXTZ bridge relies on two DVN operators to relay messages between chains. Currently, these operators are LayerZero Labs and Nethermind. The bridge's security depends on the independence of these two entities and their lack of collaboration. If the two DVNs collude or are compromised, the security of the WXTZ bridge would be at risk.

Therefore, WXTZ bridge users and WXTZ holders should carefully evaluate how much of their assets they entrust to this design. Users may feel more comfortable entrusting additional assets if more reputable DVNs are required and if the bridge owner role is secured by a multisig contract managed by multiple reputable parties.

Overview on findings

Details for each reported finding can be obtained from the “[Findings](#)” section.

Findings	Severity / Status
ELO-001: Message library not explicitly set	High / resolved
ELO-002: Not all used message libraries explicitly configured	High / resolved
ELO-003: Impossibility to identify an audited commit	High / resolved
ELO-004: No multisig configured for WXTZ owner role	High / resolved
ELO-005: DVN configuration and Etherlink commitment refutation	High / resolved Informational
ELO-006: Known security bugs in used compiler version	Low / open

Project overview

Scope

Our security assessment of the WXTZ setup for Etherlink included:

- WXTZ configuration
- WXTZ verification
- “LayerZero V2” including message libraries deployment and configuration on Etherlink
- Multisig smart contract to be used as the owner role for WXTZ.

Scope limitations

Our assessment is based on the following key assumptions and scope limitations:

- We did not review the source code of the deployed smart contracts directly; instead, we verified whether the deployed contracts are based on a codebase that has already undergone a security assessment.
- We did not evaluate the credibility, independence, or experience of the security assessors who published available security assessment reports.
- We did not assess or highlight any potential risks which may arise from the nature of the platform itself. Consequently, we did not analyse the trustworthiness or capabilities of the operating team or keyholders in privileged roles, such as contract owners.
- We did not evaluate whether the WXTZ team has adequate measures and procedures for monitoring the original codebase and relevant channels, or for responding appropriately if changes occur in the original code. It is important to note that code changes might be made in response to detected vulnerabilities.
- We relied on publicly available block explorers¹ to obtain the deployed smart contract bytecode, assuming them to be independent and capable of providing correct results.
- We did not assess the security of the following infrastructure elements, where critical vulnerabilities could potentially compromise the WXTZ:
 - a. Tezos Layer 1 node

¹ <https://explorer.etherlink.com/>, <https://etherscan.io/>, <https://arbiscan.io/>, <https://basescan.org/>, <https://bscscan.com/>

- b. Tezos smart rollup node
 - c. Etherlink kernel
 - d. Etherlink EVM node
 - e. Layer Zero Executor
 - f. Layer Zero Endpoints contracts code
 - g. Layer Zero Message Libraries contracts code
 - h. WXTZ contract code
- We did not review the WXTZ user frontend (web application), where vulnerabilities could compromise the security of user-initiated transfers if users blindly sign transactions through this interface.

Objectives

The objective is to conduct an independent security assessment of the WXTZ setup for Etherlink to evaluate its overall security. However, our assessment depends on existing security reports for involved smart contracts and does not cover several infrastructure components. Please refer to the [“Scope limitations”](#) section for details.

Activities

Here is a list of general activities conducted during the security assessment for the previously defined scope:

- General
 - a. Build understanding based on available documentation, meetings, and Q/A.
 - b. Assess the WXTZ via a holistic assessment of the system, considering the interdependencies and interactions of its components, rather than only assessing individual components in scope.
- WXTZ configuration
 - a. Verify that the deployed code for the WXTZ smart contracts is based on a codebase that has undergone a security assessment.
 - b. Review the configuration including
 - the block confirmation number
 - the number of required DVNs
 - the number of optional DVNs
 - the optional DVN threshold

- the identity of the DVNs
 - c. Verify that the multisig smart contract is assigned to the WXTZ owner role.
- WXTZ configuration
 - a. Perform a test transaction from each chain to each chain (at least 20 transactions). In-scope chains are:
 - Etherlink
 - Ethereum
 - Base
 - BNBChain
 - Arbitrum
- “LayerZero V2”
 - a. Verify that the deployed code relevant for the LayerZero V2 and its used sub contracts like libraries is based on a codebase that has undergone a security assessment.
 - b. Review the configuration relevant for the WXTZ.
- Multisig
 - a. Review the configuration.
 - b. Verify that the multisig smart contract configuration matches the expected configuration mentioned in the documentation received.
- Report on the security activities conducted, the results of the assessments, and provide actionable suggestions to improve the security of the WXTZ infrastructure

More details about our performed assessment activities can be obtained from section [“Initial assessment”](#) in the appendix.

In the reassessment conducted on January 17, 2025, we evaluated whether our reported findings, ELO-001 and ELO-002, had been appropriately resolved.

In the reassessment conducted on February 21, 2025, we evaluated whether our reported findings, ELO-004 and ELO-005, had been appropriately resolved.

In the reassessment conducted on July 15, 2025, we reviewed the newly available security assessment report from Zellic and evaluated whether it adequately addresses the issue raised in ELO-003.

Findings

ELO-001: Message library not explicitly set

If a user application such as the WXTZ doesn't specify a message library, the default one will be used. The owner of the EndPoint V2 smart contract, LayerZero Labs, can register new message libraries and change the default message library.

Thus, the current WXTZ configuration relies heavily on the correct and trustworthy behaviour of LayerZero Labs, which in turn introduces several potential risks to the WXTZ as a user application. For example:

- LayerZero Labs may fail to properly update the default libraries, resulting in assets being stuck in the WXTZ.
- LayerZero Labs could deploy an unaudited or vulnerable message library, leading to stuck or stolen assets.
- If LayerZero Labs is hacked or acts maliciously, it could steal assets being transferred through the WXTZ.

Furthermore, relying so heavily on LayerZero Labs for the proper and secure functioning of the WXTZ undermines Etherlink's promise of operating trustless bridges².

Probability - Low

Impact - High, stealing of assets transferred through the WXTZ

Severity - High

Recommendation

We strongly recommend explicitly setting the message library to be used by WXTZ.

Reassessment result

All message libraries to be used by WXTZ have been explicitly defined. Therefore, we consider this issue resolved and have updated the status from "open" to "resolved".

² <https://docs.etherlink.com/bridging/>

ELO-002: Not all used message libraries explicitly configured

The message libraries from and to Etherlink have been explicitly configured for WXTZ. However, the library configurations between other chains have not been specifically defined, so default settings are applied if the user application, WXTZ, does not provide its own configuration. For example, a transfer of WXTZ from Ethereum to Base would use the default library configuration.

LayerZero Labs, the owner of the message libraries, can modify the default settings of a remote chain configured in the library at any time.

Thus, the current WXTZ configuration relies heavily on the correct and trustworthy behaviour of LayerZero Labs, which in turn introduces several potential risks to the WXTZ as a user application. For example:

- LayerZero Labs could replace the default DVNs with untrustworthy DVN providers
- If LayerZero Labs is hacked or acts maliciously, it could alter the default library configuration, allowing messages to be injected without ever being created.

Furthermore, relying so heavily on LayerZero Labs for the proper and secure functioning of the WXTZ undermines Etherlink's promise of operating trustless bridges³.

Probability - Low

Impact - High, creating of new WXTZ

Severity - High

Recommendation

We strongly recommend explicitly configuring the chain-specific settings in the message libraries used and selecting trusted DVNs.

Reassessment result

All settings in the message libraries were explicitly configured. Therefore, we consider this issue resolved and have updated the status accordingly from "open" to "resolved".

³ <https://docs.etherlink.com/bridging/>

ELO-003: Impossibility to identify an audited commit

An issue has been identified for the following smart contracts in scope:

- LZ EndpointV2
 - Etherlink Mainnet / 0xAaB5A48CFC03Efa9cC34A2C1aAcCCB84b4b770e4
- SendUln302
 - Etherlink Mainnet / 0xc1B621b18187F74c8F6D52a6F709Dd2780C09821
- ReceiveUln302
 - Etherlink Mainnet / 0x377530cdA84DFb2673bF4d145DCF0C4D7fdcB5b6
- SendUln301
 - Etherlink Mainnet / 0x7cacBe439EaD55fa1c22790330b12835c6884a91
- ReceiveUln301
 - Etherlink Mainnet / 0x282b3386571f7f794450d5789911a9804FA346b4
- DVN Nethermind
 - Etherlink Mainnet / 0x7a23612f07d81f16b26cf0b5a4c3eca0e8668df2
- DVN LayerZero Labs
 - Etherlink Mainnet / 0xc097ab8cd7b053326dfe9fb3e3a31a0cce3b526f

While several audits⁴ have been conducted on the repository⁵ from which these smart contracts have originated, the codebase has undergone a serious restructuring that does not allow pointing to any specific audit.

Many past audits were provided with private, custom, or no-longer existing repositories. Additionally, even the ones conducted on the identified repository (see this⁶ one as of the most comprehensive in terms of scope) refer to a version that can no longer be compiled due to library and reference updates.

To reproduce the bytecode deployed on-chain, we used the latest commit available at the time of verification, version 2d4d177d017d525bfe11892a98f2c511155d2160. With this commit it is possible to reproduce the deployed bytecode. However, Inference can make no assumption on whether this commit matches a past audited version, and to what extent.

This means that it is not possible to easily reproduce the deployed bytecode and determine whether it is obtained from an audited codebase.

Probability - Unknown

⁴ <https://github.com/LayerZero-Labs/Audits>

⁵ <https://github.com/LayerZero-Labs/LayerZero-v2>

⁶ <https://github.com/LayerZero-Labs/Audits/blob/main/audits/EndpointV2-Paladin-15DEC2023.pdf>



Impact - High, with an unaudited codebase all risks are possible, including frozen, lost or stolen assets

Severity - High

Recommendation

We strongly recommend addressing the issue by identifying a past audit that can be easily compiled to originate the deployed bytecode. If such an audit is not available, we recommend conducting a new audit to reflect the structural changes undergone by the codebase.

Reassessment result:

On July 1, 2025, Zellic updated their security assessment report⁷ to reflect commit 2d4d177d. Inference reviewed the report and cross-checked the source code to verify that the issues identified in Zellic's report were addressed in the referenced commit, as Zellic's report does not clearly state whether the fixes were implemented in this commit. Our evaluation confirms that all findings detailed in chapter 4, 'Detailed Findings,' have been resolved in commit 2d4d177d017d525bfe11892a98f2c511155d2160.

Therefore, we consider ELO-003 resolved and have updated the status accordingly from "open" to "resolved".

ELO-004: No multisig configured for WXTZ owner role

The WXTZ owner role on the different chains is held by an EOA, which, to our knowledge, does not rely on an MPC solution.

This means that the individual or entity controlling the EOA address has the authority to reconfigure the WXTZ bridge solution. The owner role could reconfigure the WXTZ to use different smart contracts, potentially locking assets in custody or stealing transferred assets.

Therefore, users of the WXTZ must trust that the WXTZ owner role will act responsibly and ensure that the private keys for the EOA address are securely managed and used.

⁷ Report name: Endpoint V2 - Zellic Audit Report_250704_002213.pdf / sha256sum: c341f090c4e6033fbff85275cf290969db886f43454abb6416f32db774706487

Probability - Unknown, since we do not know by whom the EOA is controlled and how the private keys are handled.

Impact - High, locked assets or stealing of assets in transfer by WXTZ owner

Severity - High

Recommendation:

We recommend implementing a multisig smart contract solution to distribute the owner role among multiple individuals or entities. This solution should be transparently documented to allow WXTZ users to stay informed and build trust in the WXTZ ownership.

Reassessment result:

All owner roles have been transferred to a multisig solution. Therefore, we consider this issue resolved and have updated the status from "open" to "resolved".

ELO-005: DVN configuration and Etherlink commitment refutation

Etherlink is an EVM-compatible blockchain built on Tezos' enshrined optimistic rollup technology. Consequently, the security of the WXTZ bridge solution, which is based on LayerZero V2, depends also on the assumption that at least one honest Etherlink rollup operator exists.

We inquired about the configurations of the two DVNs: LayerZero Labs and Nethermind.

For DVN LayerZero Labs, we were provided with links to a public repository and an access-restricted one. As we could not access the restricted repository and the public repository did not contain any Etherlink-related information, we were unable to determine whether DVN LayerZero Labs is configured to run Etherlink setups using blueprints finalized in Tezos L1 blocks or to identify the operating mode of the Smart Rollup.

Regarding DVN Nethermind, we were directed to a subfolder in the official Tezos source code repository. It appears that Nethermind's setup is based on this repository. If no modifications have been made, the setup would operate in "Observer" mode, relying solely on blueprints finalized in Tezos L1 blocks.



Given the lack of definitive information, we assume the worst-case scenario: both DVNs are running in "Observer" mode, and DVN LayerZero Labs is not using finalized blueprints.

Probability - Medium

Impact - High, WXTZ bridge users potentially lose their assets in transfer

Severity - High

Recommendation:

We recommend that at least one designated DVN operates an Etherlink rollup capable of refuting incorrect commitments. Additionally, this should be transparently documented in the official WXTZ documentation.

At a minimum, the WXTZ documentation should include verifiable information about the entities operating Etherlink rollups, enabling WXTZ users to independently verify the existence of at least one honest Etherlink rollup operator.

Furthermore, we recommend ensuring that DVN LayerZero Labs uses only finalized blueprints. Additionally, both configured DVNs should explicitly confirm that they are using finalized blueprints as part of their setup.

Reassessment result:

The WXTZ documentation has been updated, allowing users to examine their reliance on Etherlink node operators. As a result, we consider this finding "resolved" but classify it as informational to ensure report readers are aware of it.

ELO-006: Known security bugs in used compiler version

The deployed smart contracts are compiled using version 0.8.22 of the Solidity compiler. This is an outdated version with a low severity reported security vulnerability.

As a result, we determine the overall risk to be "low," based on the category referenced in the Solidity security bulletins.

Notes: We have not evaluated the code for susceptibility to this reported vulnerability.



Recommendation:

We recommend consulting with the different code developers and operators to confirm that they have evaluated the code's susceptibility to these reported vulnerabilities. Additionally, we recommend that the WXTZ team, if not already doing so, monitor published Solidity security bulletins and evaluate their potential impact on the WXTZ.

Comment from WXTZ team:

The bug is around the verbatim builtin Yul code which is not present in the whole code base of the WXTZ so we can say that we don't risk anything on that.

Disclaimer

This security assessment report ("Report") by Inference AG ("Inference") is solely intended for the Tezos Foundation ("Client") with respect to the Report's purpose as agreed by the Client. The Report may not be relied upon by any other party than the Client and may only be distributed to a third party or published with the Client's consent. If the Report is published or distributed by the Client or Inference (with the Client's approval) then it is for information purposes only and Inference does not accept or assume any responsibility or liability for any other purpose or to any other party.

Security assessments of a software or technology cannot uncover all existing vulnerabilities. Even an assessment in which no weaknesses are found is not a guarantee of a secure system. Generally, code assessments enable the discovery of vulnerabilities that were overlooked during development and show areas where additional security measures are necessary. Within the Client's defined time frame and engagement, Inference has performed an assessment in order to discover as many vulnerabilities of the technology or software analysed as possible. The focus of the Report's security assessment was limited to the general items and code parts defined by the Client. The assessment shall reduce risks for the Client but in no way claims any guarantee of security or functionality of the technology or software that Inference agreed to assess. As a result, the Report does not provide any warranty or guarantee regarding the defect-free or vulnerability-free nature of the technology or software analysed.



In addition, the Report only addresses the issues of the system and software at the time the Report was produced. The Client should be aware that blockchain technology and cryptographic assets present a high level of ongoing risk. Given the fact that inherent limitations, errors or failures in any software development process and software product exist, it is possible that even major failures or malfunctions remain undetected by the Report. Inference did not assess the underlying third party infrastructure which adds further risks. Inference relied on the correct performance and execution of the included third party technology itself.

Appendix

Initial assessment

This section of the Appendix shows the activities and results conducted in the initial assessment phase.

WXTZ' LayerZero integration

Etherlink (LZ chainID: 30292)

No.	Description	Result
1	<p>Check which EndPoint V2 is configured at the WXTZ and check whether this is the official LayerZero V2 endpoint for Etherlink mainnet.</p> <p>Result: Configured address is: 0xAaB5A48CFC03Efa9cC34A2C1aAcCCB84b4b770e4, which is the official EndPoint V2 for Etherlink mainnet according to: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p>	Ok
2	<p>Check which send library is configured for WXTZ at EndPointV2 and check whether the library is the ULNV2 libraries registered here: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p> <p>Result: All chains in scope are using 0xc1B621b18187F74c8F6D52a6F709Dd2780C09821</p>	Ok
3	<p>Check whether the configured send library is the default one.</p> <p>Result. Default one for all chains in scope.</p>	Not ok

inference

□-□-□-□-■

4	<p>Check which receive libraries are configured for WXTZ and check whether the library is the ULNV2 libraries registered here: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p> <p>Result: All chains in scope are using: 0x377530cdA84DFb2673bF4d145DCF0C4D7fdcB5b6</p>	Ok
5	<p>Check whether the configured receive library is the default one.</p> <p>Result. Default one for all chains in scope.</p>	Not ok
6	<p>Check who is registered as the delegate.</p> <p>Result: 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05</p>	Info
7	<p>Check whether the registered delegate is a multisig contract.</p>	Not ok
8	<p>Check whether the configured send library is not using the default config.</p> <p>Result: All chains have a custom config set.</p>	Ok
9	<p>Review the configuration set for the send library.</p> <p>Result: For all chains the same configuration is set:</p> <ul style="list-style-type: none"> - Confirmation: 1 - RequiredDVNCount: 2 - OptionalDVNCount: 0 - OptionalDVNThreshold: 0 - RequiredDVNs are: 	Ok

inference

□-□-□-□-■

	<ul style="list-style-type: none"> - Nethermind/BWARE: 0x7a23612f07d81f16b26cf0b5a4c3eca0e8668df2 - LayerZero Labs: 0xc097ab8cd7b053326dfe9fb3e3a31a0cce3b526f - OptionalDVNs: [] <p>Both DVNs can be found in the official list: https://docs.layerzero.network/v2/developers/evm/technical-reference/dvn-addresses</p>	
10	<p>Check whether the configured receive library is not using the default config.</p> <p>Result: All chains have a custom config set.</p>	Ok
11	<p>Review the configuration set for the receive library.</p> <p>Result:</p> <ul style="list-style-type: none"> - Confirmation: Parameters are adjusted to the different chains. - RequiredDVNCount: 2 - OptionalDVNCount: 0 - OptionalDVNThreshold: 0 - RequiredDVNs are: <ul style="list-style-type: none"> - Nethermind/BWARE: 0x7a23612f07d81f16b26cf0b5a4c3eca0e8668df2 - LayerZero Labs: 0xc097ab8cd7b053326dfe9fb3e3a31a0cce3b526f - OptionalDVNs: [] <p>Both DVNs can be found in the official list: https://docs.layerzero.network/v2/developers/evm/technical-reference/dvn-addresses</p>	Ok

12	<p>Check whether the confirmation settings for the receive library have not been lowered in the WXTZ setup, but correspond to at least LZ's default settings.</p> <p>Result: All confirmation values are equal to LZ's default ones for the specific chain.</p>	Ok
----	---	-----------

ArbitrumOne (LZ chainID: 30110)

No.	Description	Result
1	<p>Check which EndPoint V2 is configured at the WXTZ and check whether this is the official LayerZero V2 endpoint for this chain.</p> <p>Result: Configured address is: 0x1a44076050125825900e736c501f859c50fE728c a, which is the official EndPoint V2 for Etherlink mainnet according to: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p>	Ok
2	<p>Check which send library is configured for WXTZ at EndPointV2 and check whether the library is the ULNV2 libraries registered here: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p> <p>Result: All chains in scope are using 0xc1B621b18187F74c8F6D52a6F709Dd2780C09821</p>	Ok
3	<p>Check whether the configured send library is the default one.</p> <p>Result.</p>	Not ok

inference

□-□-□-□-■

	Default one for all chains in scope.	
4	<p>Check which receive libraries are configured for WXTZ and check whether the library is the ULNV2 libraries registered here: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p> <p>Result: All chains in scope are using: 0x377530cdA84DFb2673bF4d145DCF0C4D7fdcB5b6</p>	Ok
5	<p>Check whether the configured receive library is the default one.</p> <p>Result. Default one for all chains in scope.</p>	Not ok
6	<p>Check who is registered as the delegate.</p> <p>Result: 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05</p>	Info
7	Check whether the registered delegate is a multisig contract.	Not ok
8	<p>Check whether the configured send library is not using the default config.</p> <p>Result: Only the send library for Etherlink does not use the default. All other connections are using the default.</p>	Not ok
9	Review the configuration set for the send library.	Ok
10	<p>Check whether the configured receive library is not using the default config.</p> <p>Result:</p>	Not ok

inference

□-□-□-□-■

	Only the receive library for Etherlink does not use the default. All other connections are using the default.	
11	Review the configuration set for the receive library.	Ok
12	<p>Check whether the confirmation settings for the receive library have not been lowered in the WXTZ setup, but correspond to at least LZ's default settings.</p> <p>Result: All confirmation values are equal to LZ's default ones for the specific chain.</p>	Ok

Base (LZ chainID: 30184)

No.	Description	Result
1	<p>Check which EndPoint V2 is configured at the WXTZ and check whether this is the official LayerZero V2 endpoint for this chain.</p> <p>Result: Configured address is: 0x1a44076050125825900e736c501f859c50fE728c a, which is the official EndPoint V2 for Etherlink mainnet according to: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p>	Ok
2	<p>Check which send library is configured for WXTZ at EndPointV2 and check whether the library is the ULNV2 libraries registered here: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p> <p>Result:</p>	Ok

inference

□-□-□-□-■

	All chains in scope are using 0xc1B621b18187F74c8F6D52a6F709Dd2780C09821	
3	Check whether the configured send library is the default one. Result. Default one for all chains in scope.	Not ok
4	Check which receive libraries are configured for WXTZ and check whether the library is the ULNV2 libraries registered here: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts Result: All chains in scope are using: 0x377530cdA84DFb2673bF4d145DCF0C4D7fdcB5b6	Ok
5	Check whether the configured receive library is the default one. Result. Default one for all chains in scope.	Not ok
6	Check who is registered as the delegate. Result: 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05	Info
7	Check whether the registered delegate is a multisig contract.	Not ok
8	Check whether the configured send library is not using the default config. Result: Only the send library for Etherlink does not use the default. All other connections are using the default.	Not ok
9	Review the configuration set for the send library.	Ok

10	<p>Check whether the configured receive library is not using the default config.</p> <p>Result: Only the receive library for Etherlink does not use the default. All other connections are using the default.</p>	Not ok
11	Review the configuration set for the receive library.	Ok
12	<p>Check whether the confirmation settings for the receive library have not been lowered in the WXTZ setup, but correspond to at least LZ's default settings.</p> <p>Result: All confirmation values are equal to LZ's default ones for the specific chain.</p>	Ok

BNB Smart Chain (LZ chainID: 30102)

No.	Description	Result
1	<p>Check which EndPoint V2 is configured at the WXTZ and check whether this is the official LayerZero V2 endpoint for this chain.</p> <p>Result: Configured address is: 0x1a44076050125825900e736c501f859c50fE728c a, which is the official EndPoint V2 for Etherlink mainnet according to: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p>	Ok
2	<p>Check which send library is configured for WXTZ at EndPointV2 and check whether the library is the ULNV2 libraries registered here: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p>	Ok

inference

□-□-□-□-■

	<p>Result:</p> <p>All chains in scope are using 0xc1B621b18187F74c8F6D52a6F709Dd2780C09821</p>	
3	<p>Check whether the configured send library is the default one.</p> <p>Result.</p> <p>Default one for all chains in scope.</p>	Not ok
4	<p>Check which receive libraries are configured for WXTZ and check whether the library is the ULNV2 libraries registered here: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p> <p>Result:</p> <p>All chains in scope are using: 0x377530cdA84DFb2673bF4d145DCF0C4D7fdcB5b6</p>	Ok
5	<p>Check whether the configured receive library is the default one.</p> <p>Result.</p> <p>Default one for all chains in scope.</p>	Not ok
6	<p>Check who is registered as the delegate.</p> <p>Result:</p> <p>0x21c79736B62A0C9a1c843D9F99049Bac391B9A05</p>	Info
7	<p>Check whether the registered delegate is a multisig contract.</p>	Not ok
8	<p>Check whether the configured send library is not using the default config.</p> <p>Result:</p> <p>Only the send library for Etherlink does not use the default. All other connections are using the default.</p>	Not ok

9	Review the configuration set for the send library.	Ok
10	<p>Check whether the configured receive library is not using the default config.</p> <p>Result: Only the receive library for Etherlink does not use the default. All other connections are using the default.</p>	Not ok
11	Review the configuration set for the receive library.	Ok
12	<p>Check whether the confirmation settings for the receive library have not been lowered in the WXTZ setup, but correspond to at least LZ's default settings.</p> <p>Result: All confirmation values are equal to LZ's default ones for the specific chain.</p>	Ok

Ethereum (LZ chainID: 30101)

No.	Description	Result
1	<p>Check which EndPoint V2 is configured at the WXTZ and check whether this is the official LayerZero V2 endpoint for this chain.</p> <p>Result: Configured address is: 0x1a44076050125825900e736c501f859c50fE728c a, which is the official EndPoint V2 for Etherlink mainnet according to: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p>	Ok
2	<p>Check which send library is configured for WXTZ at EndPointV2 and check whether the library is the ULNV2 libraries registered here: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p>	Ok

inference

□-□-□-□-■

	<p>Result:</p> <p>All chains in scope are using</p> <p>0xc1B621b18187F74c8F6D52a6F709Dd2780C09821</p>	
3	<p>Check whether the configured send library is the default one.</p> <p>Result.</p> <p>Default one for all chains in scope.</p>	Not ok
4	<p>Check which receive libraries are configured for WXTZ and check whether the library is the ULNV2 libraries registered here: https://docs.layerzero.network/v2/developers/evm/technical-reference/deployed-contracts</p> <p>Result:</p> <p>All chains in scope are using:</p> <p>0x377530cdA84DFb2673bF4d145DCF0C4D7fdcB5b6</p>	Ok
5	<p>Check whether the configured receive library is the default one.</p> <p>Result.</p> <p>Default one for all chains in scope.</p>	Not ok
6	<p>Check who is registered as the delegate.</p> <p>Result:</p> <p>0x21c79736B62A0C9a1c843D9F99049Bac391B9A05</p>	Info
7	<p>Check whether the registered delegate is a multisig contract.</p>	Not ok
8	<p>Check whether the configured send library is not using the default config.</p> <p>Result:</p>	Not ok

	Only the send library for Etherlink does not use the default. All other connections are using the default.	
9	Review the configuration set for the send library.	Ok
10	<p>Check whether the configured receive library is not using the default config.</p> <p>Result: Only the receive library for Etherlink does not use the default. All other connections are using the default.</p>	Not ok
11	Review the configuration set for the receive library.	Ok
12	<p>Check whether the confirmation settings for the receive library have not been lowered in the WXTZ setup, but correspond to at least LZ's default settings.</p> <p>Result: All confirmation values are equal to LZ's default ones for the specific chain.</p>	Ok

WXTZ configuration

Etherlink (LZ chainID: 30292)

No.	Description	Result
1	<p>Check who is the owner of the WXTZ smart contract.</p> <p>Result: 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05</p>	Info
2	Check whether the owner is a multisig contract.	Not ok

inference



3	Obtain configured peers and check whether they match with remote WXTZ contracts for the specific chains.	Ok
---	--	----

ArbitrumOne (LZ chainID: 30110)

No.	Description	Result
1	Check who is the owner of the WXTZ smart contract. Result: 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05	Info
2	Check whether the owner is a multisig contract.	Not ok
3	Obtain configured peers and check whether they match with remote WXTZ contracts for the specific chains.	Ok

Base (LZ chainID: 30184)

No.	Description	Result
1	Check who is the owner of the WXTZ smart contract. Result: 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05	Info
2	Check whether the owner is a multisig contract.	Not ok
3	Obtain configured peers and check whether they match with remote WXTZ contracts for the specific chains.	Ok

BNB Smart Chain (LZ chainID: 30102)

No.	Description	Result
1	Check who is the owner of the WXTZ smart contract. Result: 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05	Info
2	Check whether the owner is a multisig contract.	Not ok

inference



3	Obtain configured peers and check whether they match with remote WXTZ contracts for the specific chains.	Ok
---	--	----

Ethereum (LZ chainID: 30101)

No.	Description	Result
1	Check who is the owner of the WXTZ smart contract. Result: 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05	Info
2	Check whether the owner is a multisig contract.	Not ok
3	Obtain configured peers and check whether they match with remote WXTZ contracts for the specific chains.	Ok

DVNs

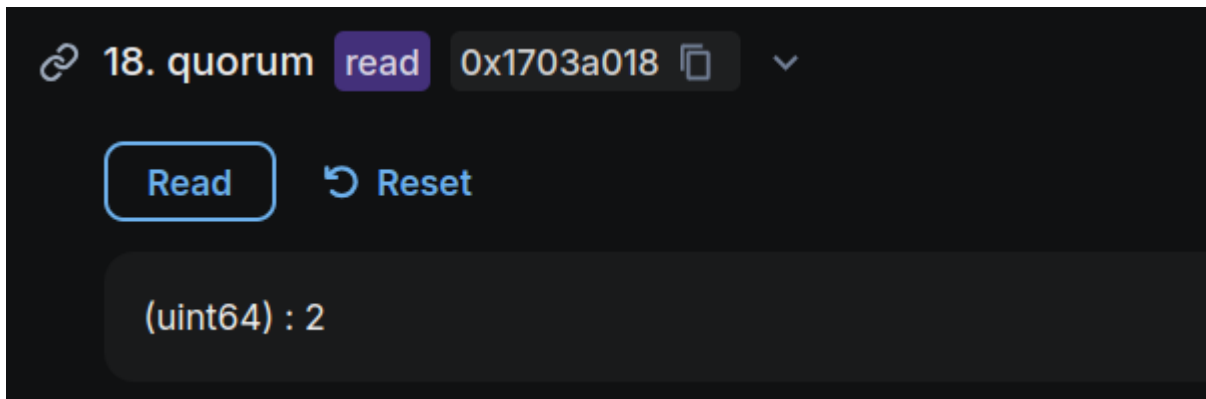
LayerZero Labs

No.	Description	Result
1	Inquire/check what the verification model is for this DVN. Result: The DVN LayerZero is relaying messages as soon as the required block confirmations have been reached.	Info
2	Check what the configured quorum is. Result: Quorum =2	Info
3	Check whether the DVN infrastructure is configured to only consider finalised L1 blocks. Result: No information received.	Not ok
4	Check/Inquire what smart rollup mode their Etherlink instance is running. Result: No information received.	Not ok

Details #2

https://explorer.etherlink.com/address/0xc097ab8CD7b053326DFe9fB3E3a31a0CCe3B526f?tab=read_write_contract

inference



Nethermind

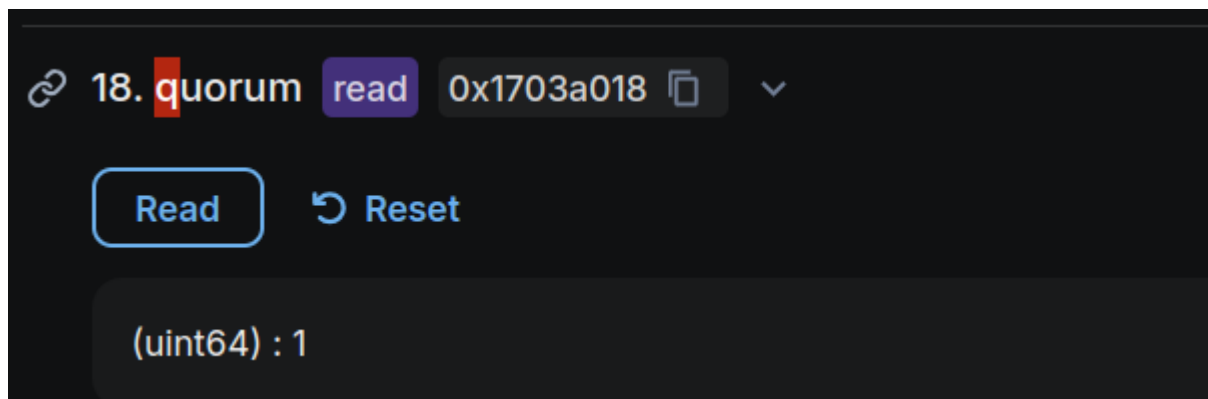
No.	Description	Result
1	Inquire/check what the verification model is for this DVN. Result: The DVN Nethermind is relaying messages as soon as the required block confirmations have been reached.	Info
2	Check what the configured quorum is. Result: Quorum =1	Info
3	Check whether the DVN infrastructure is configured to only consider finalised L1 blocks. Result: We received the following config: https://gitlab.com/tezos/tezos/-/tree/90b7bcc46ca40403f151ad865277b61d7397d823/etherlink/scripts/docker-compose/mainnet-docker-compose	Ok

inference

	Assuming they are really using this configuration that would be fine.	
4	<p>Check/Inquire what smart rollup mode their Etherlink instance is running.</p> <p>Result: We received the following config: https://gitlab.com/tezos/tezos/-/tree/90b7bcc46ca40403f151ad865277b61d7397d823/etherlink/scripts/docker-compose/mainnet-docker-compose</p> <p>Assuming they are really using this configuration that would be fine.</p>	Ok

Details #2

https://explorer.etherlink.com/address/0x7a23612F07d81F16B26cF0b5a4C3eca0E8668df2?tab=read_write_contract



Multisig setup

No.	Description	Result
1	Check whether the relevant roles for WXTZ are controlled using a multisig contract.	Not Ok

	<p>Result:</p> <p>The owner for the different contracts is <code>0xC3A9d37F723b06feB1Df317b10E39DD2dE699369</code>, an EOA address. Based on our inquiry there is no MPC solution in place.</p>	
2	<p>Inquire information about the planned multisig setup to be used.</p> <p>Result:</p> <p>The setup will be a Safe multisig.</p>	Info
3	<p>Check whether the multisig setup is based on a security reviewed code base.</p> <p>Results:</p> <p>See below.</p>	Ok
4	<p>Check whether the multisig setup is correctly configured.</p> <p>Results:</p> <p>See below.</p>	Not ok

Details #3

The multisig setup was verified against the audited versions 1.3.0 for Etherlink Mainnet⁸, and 1.4.1 for the EVM-based blockchains Ethereum, Arbitrum, Base, BNB Smart Chain⁹. The deployed bytecode uses the official factory provided by Gnosis, and all bytecodes rely on expected addresses of the versions in use, as reported in the official changelog from Gnosis¹⁰. The data supplied to the Factory was decompiled and analysed to verify the deployment consisted of standard and expected procedures and values.

⁸ Audits available for 1.3.0:

https://github.com/safe-global/safe-smart-account/blob/main/docs/audit_1_3_0.md

⁹ Audits available for 1.4.1:

https://github.com/safe-global/safe-smart-account/blob/main/docs/audit_1_4_0.md

¹⁰ <https://github.com/safe-global/safe-smart-account/blob/main/CHANGELOG.md>

Details #4

The configuration of the multisig solution includes an additional party as representative address, namely **0x146F2D294aAE7996Bd95bae526472f751D32258c**

The documentation provided to Inference at the time of the assessment reports that this address will be replaced with a new one owned by a known party. However, no knowledge of this new address nor when the expected change should happen has been communicated.

Additionally, the configured threshold reported is 1 out of 5, instead of the reported 3 out of 5. This issue is present on the following chains:
















- Etherlink Mainnet
- Ethereum
- Arbitrum
- Base
- BNB Chain

The following screenshot shows the configuration on Etherlink mainnet.

Manage Safe Account signers

Add, remove and replace or rename existing signers. Signer names are only stored locally and will never be shared with us or any third parties.

Name

 etherlink:0x146F2D294aAE7996Bd95bae526472f751D32258c	 
 etherlink:0xD42CCc80AE675D957324f8cBB7cB7FD1cD2D96DD	 
 etherlink:0xb0DB59274a6c2837a85e4e99556b099923d58c57	 
 etherlink:0xCf0289Ca48Bf8F4E28e37AAbDD16b3F1b2aD8	 
 etherlink:0x542DFCC8a11C7bEB667A997fd9b4858c5557D73e	 

+ Add new signer [Export as CSV](#)

Required confirmations

Any transaction requires the confirmation of:

1 out of 5 signers. [Change](#)

inference

As an additional example, this is the same configuration, on the Ethereum mainnet:






















Members

Signers

Signers have full control over the account, they can propose, sign and execute transactions, as well as reject them.

+ Add signer

Export as CSV

	eth:0x146F2D294aAE7996Bd95bae526472f751D32258c  	  
	raidsquare.eth  	  
	eth:0xb0DB59274a6c2837a85e4e99556b099923d5Bc57  	  
	eth:0xCf02B9Ca488f8F6F4E28e37AA1bDD16b3F1b2aD8  	  
	eth:0x542DFCC8a11C7bEB667A997fd9b485Bc5557D73e  	  

Proposers New

Proposers can suggest transactions but cannot approve or execute them. Signers should review and approve transactions first. [Learn more](#)

+ Add proposer

Required confirmations

Any transaction requires the confirmation of:

1 out of 5 signers. Change

Deployed bytecode verification

No.	Description	Result
1	Check whether the creation and deployment bytecode on-chain are based on previously-audited smart contracts. In case the source code differs from the originally audited source code, evaluate whether the changes could introduce security vulnerabilities.	Not ok

	<p>Result:</p> <p>Not all the source code matches previously audited versions. Thus, for part of the codebase it is not possible to determine whether, and to what extent, the source code has been audited and reviewed.</p> <p>See additional details.</p>	
2	<p>Check whether the creation and deployed bytecode on-chain match the ones obtained from compiling the smart contracts from the original source code.</p> <p>Result:</p> <p>The bytecodes match the reference source code. The only difference is to be found in constructor arguments.</p> <p>This does not pose a security risk. The bytecodes often include references to other smart contract addresses and parameters which cannot 100% match between different deployments, as they will need to point to different instances of the same contract. Constructor arguments have been checked to verify that smart contracts pointed to known, verified addresses.</p>	Ok
3	<p>Check whether the addresses marked as EOA are actually EOAs and do not need a bytecode verification.</p> <p>Result:</p> <p>Some addresses that were present in the asset list given to Inference regarding WXTZ were excluded from the verification activity. The verification was not performed because the following addresses are EOAs and not smart contracts and, as such, they have no source code to be verified.</p>	Ok

Details #1

Perfectly matching an audited version:

- Wrapped XTZ (WXTZ)
 - Etherlink Mainnet / 0xc9B53AB2679f573e480d01e0f49e2B5CFB7a3EAb
 - Arbitrum / 0x7424f00845777A06E21F0bd8873f814A8A814B2D



- Base / 0x91F9cc2649ac70a071602cadE9b0C1A5868af51D
- BNB Smart Chain / 0x91F9cc2649ac70a071602cadE9b0C1A5868af51D
- Ethereum / 0xc9B53AB2679f573e480d01e0f49e2B5CFB7a3EAb

The most recent audited version¹¹ is from this audit report¹². The source code for this smart contract has not changed from the last audited version, up to the most recent commit at the time of this verification, with commit hash **b8e0ca43554d47ea551ea2a99a2e8c7e798f3f6c**.

Impossible to identify past audits:

- LZ EndpointV2
 - Etherlink Mainnet / 0xAaB5A48CFC03Efa9cC34A2C1aAcCCB84b4b770e4
- SendUln302
 - Etherlink Mainnet / 0xc1B621b18187F74c8F6D52a6F709Dd2780C09821
- ReceiveUln302
 - Etherlink Mainnet / 0x377530cdA84DFb2673bF4d145DCF0C4D7fdcB5b6
- SendUln301
 - Etherlink Mainnet / 0x7cacBe439EaD55fa1c22790330b12835c6884a91
- ReceiveUln301
 - Etherlink Mainnet / 0x282b3386571f7f794450d5789911a9804FA346b4
- DVN Nethermind
 - Etherlink Mainnet / 0x7a23612f07d81f16b26cf0b5a4c3eca0e8668df2
- DVN LayerZero Labs
 - Etherlink Mainnet / 0xc097ab8cd7b053326dfe9fb3e3a31a0cce3b526f

While several audits¹³ have been conducted on the repository¹⁴, the codebase has undergone a serious restructuring that does not allow pointing to any specific audit. Many past audits were provided with private, custom, or no-longer existing repositories. Even the ones conducted on the identified repository (see this¹⁵ for one of the most comprehensive in terms of scope) refer to a version that can no longer be compiled due to library and reference updates.

To reproduce the bytecode deployed on-chain, we used the latest commit available at the time of verification, version **2d4d177d017d525bfe11892a98f2c511155d2160**. With this commit it is possible to reproduce the deployed bytecode.

However, Inference can make no assumption on whether this commit matches a past

¹¹

<https://github.com/etherlinkcom/token-deployments/tree/7da346c456666632ba6cf5d577fa471272c46e74>

¹² <https://omniscia.io/reports/etherlink-cross-chain-token-665c8ac479e20900180f383b/>

¹³ <https://github.com/LayerZero-Labs/Audits>

¹⁴ <https://github.com/LayerZero-Labs/LayerZero-v2>

¹⁵ <https://github.com/LayerZero-Labs/Audits/blob/main/audits/EndpointV2-Paladin-15DEC2023.pdf>

audited version, and to what extent.

Details #2

Contract Name	Contract Address (compiler version used)	Result of the verification process
WXTZ	0xc9B53AB2679f573e480d01e0f49e2B5CFB7a3EAb (v0.8.22)	Ok The creation and deployment bytecodes generated from compiling the source code perfectly match the published ones.
LZ Endpoint V2	0xAaB5A48CFC03Efa9cC34A2C1aAcCCB84b4b770e4 (v0.8.22)	Ok Creation and deployment bytecodes match (they differ for constructor arguments).
SendUln302	0xc1B621b18187F74c8F6D52a6F709Dd2780C09821 (v0.8.22)	Ok Creation and deployment bytecodes match (they differ for constructor arguments).
ReceiveUln302	0x377530cdA84DFb2673bF4d145DCF0C4D7fdcB5b6 (v0.8.22)	Ok Creation and deployment bytecodes match (they differ for constructor arguments).
SendUln301	0x7cacBe439EaD55fa1c22790330b12835c6884a91 (v0.8.22)	Ok Creation and deployment bytecodes match (they differ for constructor arguments).
ReceiveUln301	0x282b3386571f7f794450d5789911a9804FA346b4 (v0.8.22)	Ok Creation and deployment bytecodes match (they differ for constructor arguments).
DVN Nethermind	0x7a23612f07d81f16b26cf0b5a4c3eca0e8668df2	Ok Creation and deployment

inference

□-□-□-□-■

	(v0.8.22)	bytecodes match (they differ for constructor arguments).
DVN LayerZero Labs	0xc097ab8cd7b053326dfe9fb3e3a31a0cce3b526f (v0.8.22)	Ok Creation and deployment bytecodes match (they differ for constructor arguments).
WXTZ (Arbitrum)	0x7424f00845777A06E21F0bd8873f814A8A814B2D (v0.8.22)	Ok The creation and deployment bytecodes generated from compiling the source code perfectly match the published ones.
WXTZ (Base)	0x91F9cc2649ac70a071602cadE9b0C1A5868af51D (v0.8.22)	Ok The creation and deployment bytecodes generated from compiling the source code perfectly match the published ones.
WXTZ (BNB)	0x91F9cc2649ac70a071602cadE9b0C1A5868af51D (v0.8.22)	Ok The creation and deployment bytecodes generated from compiling the source code perfectly match the published ones.
WXTZ (Ethereum)	0xc9B53AB2679f573e480d01e0f49e2B5CFB7a3EAb (v0.8.22)	Ok The creation and deployment bytecodes generated from compiling the source code perfectly match the published ones.

Details #3

List of EOAs outscoped from bytecode verification:

- Owner WXTZ
 - Etherlink Mainnet / 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05
 - Arbitrum / 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05
 - Base / 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05

- BNB Smart Chain / 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05
- Ethereum / 0x21c79736B62A0C9a1c843D9F99049Bac391B9A05
- Owner LZ Endpoint V2
 - Etherlink Mainnet / 0x9F403140Bc0574D7d36eA472b82DAa1Bbd4eF327

Solidity compiler version

No.	Description	Result
1	<p>Check whether the used Solidity compiler version 0.8.22 has known vulnerabilities.</p> <p>See details below.</p>	Not ok

Details #1

0.8.23	<p>https://soliditylang.org/blog/2023/11/08/solidity-0.8.23-release-announcement</p> <ul style="list-style-type: none"> ● Details: https://soliditylang.org/blog/2023/11/08/verbatim-invalid-deduplication-bug/ ● Bug severity: low ● The bug existed since version 0.8.5 <p>Potentially relevant depending on implementation code.</p>	Not ok
--------	---	---------------

Bridge solution verification

No.	Description	Result
-----	-------------	--------

inference



1	<p>Check whether the solution allows to bridge WXTZ across all in-scope blockchains.</p> <p>The in-scope blockchains are:</p> <ul style="list-style-type: none">• Etherlink• Ethereum• BNB Chain• Base• Arbitrum <p>Result:</p> <p>All transfers have completed successfully, and it has been possible to bridge WXTZ across all chains. All 20 possible transfers across the 5 chains have been tested. For further details, see Details #1 below.</p>	Ok
---	---	----

Details #1

From	To	LayerZero Scan Explorer	Evaluation
Etherlink	Arbitrum	https://layerzeroscan.com/tx/0x79f3a23ca5435348b34be34f9f538d17caef494bbe5b8c95166dd9ba5ad4776	Transfer completed successfully
Arbitrum	Base	https://layerzeroscan.com/tx/0xf6f2c1dd4a24783d4e5a17136d1c04ab9401856cef4781712b0f6d674c994cb6	Transfer completed successfully
Base	Arbitrum	https://layerzeroscan.com/tx/0x970ed63b4419fd69f086fec2263e1bff8eb99063a5344a9d626f0df7274a56fa	Transfer completed successfully
Arbitrum	BSC	https://layerzeroscan.com/tx/0x970ed63b4419fd69f086fec2263e1bff8eb99063a5344a9d626f0df7274a56fa	Transfer completed

inference



		n.com/tx/0x3d7c899e1ab1bb4afefef2f9e1864982bc7cf615baeb4947d620dc14d1053cd2	successfully
BSC	Arbitrum	https://n.com/tx/0xcd4be9899f20a4f7e1fafc540c2b5b56f4dd88c2b062c5c65888866f5e689b6	Transfer completed successfully
Arbitrum	Ethereum	https://n.com/tx/0xf5c6ff13bca24b87c0f2c71113b391cb4933d4e7cd2e6cf031030dc954d277c3	Transfer completed successfully
Arbitrum	Etherlink	https://n.com/tx/0xe57219394c278a751111bb92f2c009c8204f2d74d9c029eb9757ee2b32fe943a	Transfer completed successfully
Etherlink	Base	https://n.com/tx/0x1b80bb02a5f4079bd3c40a3c62f4bb5aebf7feb74eb19a602a3d8683bf47712e	Transfer completed successfully
Base	BSC	https://n.com/tx/0x34f18121b22376c8b10c513e6b5256842001a8fd31b0c64e2d4ab065e883e192	Transfer completed successfully
BSC	Base	https://n.com/tx/0xc1d7ed10ea59716a3821e5a03e262caf263c55514299cd5fa322b814c	Transfer completed successfully

inference



		d0d8268	
Base	Etherlink	https://layerzeroscan.com/tx/0x8ee797fb7f6e6ce39e8335934bcceb4dbcc07fe9ad23d01905dfbbf78b88cbf	Transfer completed successfully
Etherlink	BSC	https://layerzeroscan.com/tx/0x5006b9b387d94c82b376f5ed30a025baf9f53c23f063c57826aed4667190cc1a	Transfer completed successfully
BSC	Etherlink	https://layerzeroscan.com/tx/0x03290f2f73c60ed401a5d7d123946f114815aa7609f60eee87a43c8e21627c76	Transfer completed successfully
Ethereum	Arbitrum	https://layerzeroscan.com/tx/0x1e14bd4074f55bcecf127a36fd7bf528b33f57ac76dd292dac7424c8199d064	Transfer completed successfully
Base	Ethereum	https://layerzeroscan.com/tx/0x96774657feb218358d07bff2608db73fdfa6fdff38d0035b2b717cf057f4e18d	Transfer completed successfully
Ethereum	Base	https://layerzeroscan.com/tx/0x11db69b5655269472c3fa36930959a546414d64a25ab183efd3eff94d841efcb	Transfer completed successfully
BSC	Ethereum	https://layerzeroscan.com/tx/0xb3d73dd	Transfer completed successfully

inference

□-□-□-□-■

		498f9a0a866d4653e1446bc3b60ba25ed2621f8c91c4fe455f879a4c8	
Ethereum	BSC	https://layerzeroscan.com/tx/0x5d4d7928a21474e37b7e2c8926acbb687b001a08a4feb8fd2547124943e9e344	Transfer completed successfully
Etherlink	Ethereum	https://layerzeroscan.com/tx/0x9577d02603033c37a779dbc3401aac3b9701d29f4d2689d9efa4a05619f3f221	Transfer completed successfully
Ethereum	Etherlink	https://layerzeroscan.com/tx/0x76cdc203084e5d647869d25a7a62e518dcdbb51832c03444966f1d12d08c8520	Transfer completed successfully

Reassessment

This section of the Appendix shows the activities conducted in the reassessment phase, to evaluate whether the remediations applied effectively fix the issues identified in the assessment phase.

Message library configuration

For the reassessment of ELO-001 and ELO-002, we examined whether WXTZ explicitly defines the message library and its configuration. Based on the collected configuration, we confirmed that WXTZ has clearly defined the message libraries and their configurations.

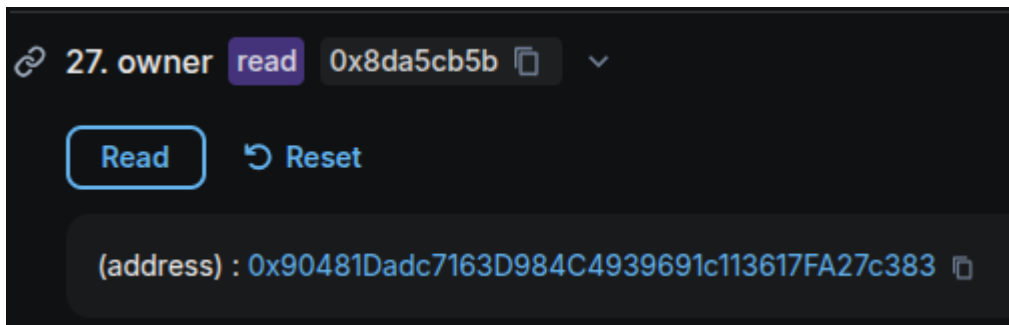
Multisig setup

No.	Description	Result
1	<p>Check whether the relevant roles for WXTZ are controlled using a multisig contract.</p> <p>Result:</p> <p>Result:</p> <p>The owner for the WXTZ contracts are:</p> <ul style="list-style-type: none"> • Etherlink Mainnet: 0x90481Dadc7163D984C4939691c113617FA27c383 • Arbitrum: 0xc22D21612462d01E4a855973609eE9AC6183B195 • Base: 0xc22D21612462d01E4a855973609eE9AC6183B195 • BNBChain: 0xc22D21612462d01E4a855973609eE9AC6183B195 • Ethereum: 0xc22D21612462d01E4a855973609eE9AC6183B195 <p>All contracts are multisig contracts.</p>	Ok
2	<p>Check whether the multisig setup is correctly configured.</p> <p>Results:</p> <p>The setup matches the configuration provided as part of the documentation to the auditing team, with a $\frac{2}{3}$ threshold and signer addresses matching the ones declared in the provided documentation.</p>	Ok

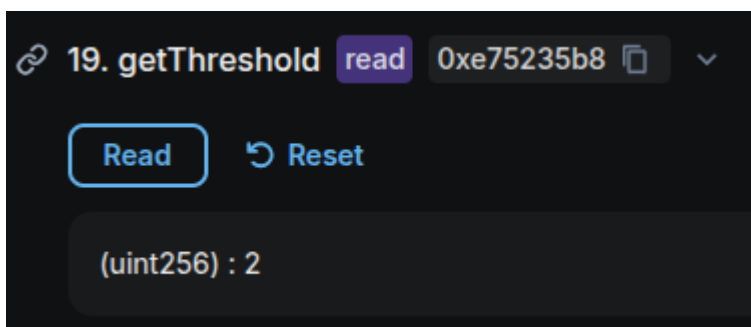
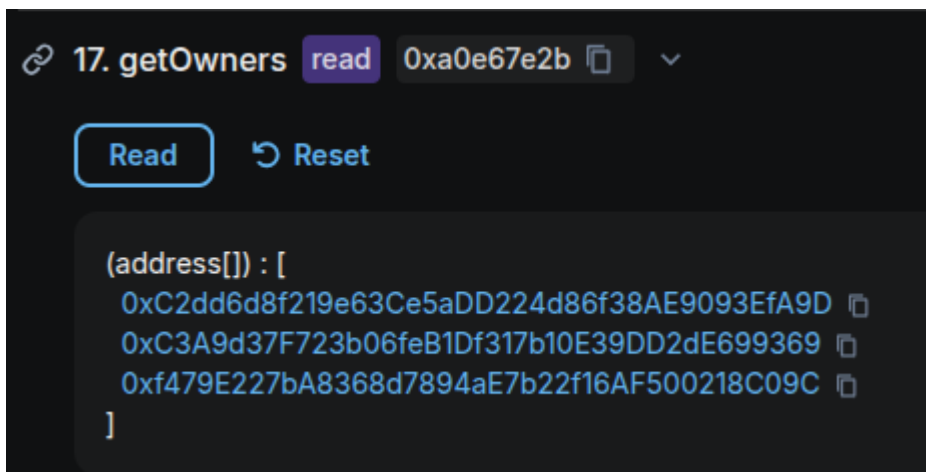
Details #1, #2

Owner of the WXTZ contract on Etherlink:

inference



The owner is a GnosisSafe Proxy, with the following registered owners and configured threshold:



Owner of the WXTZ contract on Arbitrum, Base, BSC and Ethereum:

25. owner (0x8da5cb5b)

Returns the address of the current owner.

[0xc22D21612462d01E4a855973609eE9AC6183B195](#) address



Configuration of the multisig on Arbitrum, Base, BSC and Ethereum (screenshot only captured for Arbitrum, all mentioned chains share the same configuration):

Signers





[OxC3A9d37F723b06feB1Df317b10E39DD2dE699369](#)  



[OxC2dd6d8f219e63Ce5aDD224d86f38AE9093EfA9D](#)  



[Oxf479E227bA8368d7894aE7b22f16AF500218C09C](#)  

Threshold

2 out of 3 signers

DVN configuration and Etherlink commitment refutation

The screenshot displays the Etherlink Docs website. On the left is a sidebar with navigation links: Etherlink, Status, Developers, GitLab, Get Started, Tutorial, and Bridging. The 'Bridging' section is expanded, showing sub-links like Bridging tokens, Bridging to Tezos, Bridging to EVM networks (highlighted), Bridging FA tokens, How bridging FA tokens works, and Sending FA bridging transactions. The main content area features the 'Bridge security' section, which explains that the bridge uses decentralized smart contracts from LayerZero and relies on Etherlink nodes and sequencer operators. Below this is the 'Using the EVM bridge' section, which includes a video player for 'Bridging Tokens Tutorial'. The video player shows a play button, a duration of 5 minutes, and 203 views. A Loom watermark is visible at the bottom of the video player. On the right side of the page is a 'Page Outline' section listing the document's structure: Bridge security, Using the EVM bridge, Troubleshooting, Token addresses (with sub-items for Etherlink, Ethereum, Arbitrum One, Base, Optimism, BNB Chain, and Avalanche C-Chain), How bridging wrapped assets works, and Related tools and information.

inference

The screenshot shows the Etherlink Docs website. The header includes the Etherlink logo, a search bar, and social media icons. The left sidebar contains a navigation menu with links to Etherlink, Status, Developers, GitLab, Get Started, Tutorial, Bridging, Developing, and a Network section. The Network section is expanded, showing links to Etherlink architecture, Fee structure, Network operators (highlighted), Running an Etherlink Smart Rollup node, Running an Etherlink EVM node, Building the Etherlink kernel, and Monitoring Etherlink nodes. The main content area is titled 'Network operators' and includes a sub-header 'Smart Rollup node operators'. The text explains that anyone can run Etherlink Smart Rollup nodes and that honest operators keep the network secure. It lists three organizations: The Tezos Foundation, MIDL.dev, and Zeeve. A link to the TzKT block explorer is provided to look up current Smart Rollup node operators.

Etherlink Docs

Q Search K

Network

Network operators

Etherlink relies on operators who run nodes. For information on the roles of these different nodes, see [Etherlink architecture](#).

Smart Rollup node operators

Anyone can run Etherlink [Smart Rollup nodes](#), and due to the optimistic nature of Tezos [Smart Rollups](#) it takes only one honest Smart Rollup node operator to keep Etherlink secure. Honest Smart Rollup nodes can catch any misbehavior by other nodes by refuting their incorrect commitments.

These organizations currently run Etherlink Smart Rollup nodes in operator mode to post and defend commitments for the current state of Etherlink:

- [The Tezos Foundation](#)
- [MIDL.dev](#)
- [Zeeve](#)

You can look up the current Smart Rollup node operators by checking the accounts that currently have bond set so they can post commitments for the Etherlink Smart Rollup, such as on the TzKT block explorer:
<https://tzkt.io/sr1Ghq66tYK9y3r8CC1Tf8i8m5nxb8nTvZEF/bondholders>

Page Outline

- Smart Rollup node operators
- Sequencer

Owner / delegate permission

Functions

An overview of privileged roles and their permissions to execute restricted smart contract functions.

WXTZ owner	
<ul style="list-style-type: none"> ■ renounceOwnership ■ setDelegate ■ setEnforcedOptions 	<ul style="list-style-type: none"> ■ setMsgInspector ■ setPeer (time locked) ■ setPreCrime
OAPP smart contract / delegate	
<ul style="list-style-type: none"> ■ burn ■ clear ■ nilify ■ setConfig 	<ul style="list-style-type: none"> ■ setSendLibrary ■ setReceiveLibrary ■ setReceiveLibraryTimeout ■ skip
OAPP smart contract	
<ul style="list-style-type: none"> ■ setDelegate (on LZ endpoint) 	

Risks

The following is a non-exhaustive list of high-risk scenarios that a malicious or compromised WXTZ owner, assumingly the account having WXTZ ownership permission is also the registered delegate, could potentially execute:

- Reconfiguring the bridge:
The owner could reconfigure the bridge to utilize a self-deployed and controlled smart contract on a non-Etherlink chain. This fraudulent contract could then bridge counterfeit tokens to Etherlink, facilitating the theft of legitimate XTZ locked in custody by the WXTZ smart contract. Conversely, this setup could enable the creation of new wrapped WXTZ without proper backing of XTZ in Etherlink.

inference



The time lock (2 days) implemented in setPeer may help address specific scenarios, provided the initial setPeer call is detected and acted on in a timely manner.

- Changing DVNs:

The owner could configure the WXTZ bridge solution to use different DVN operators. With operators under their control, the owner could compromise the WXTZ bridge.

- Disabling bridge:

Reconfigure the bridge parameters in such a way that the bridge becomes non-operational, resulting in assets either in transfer or in custody being potentially locked within the bridge.

Risk rating definition

Severity definition is based on the [OWASP Risk Rating Methodology](#) and [NIST SP800-30](#). It implements a categorization mechanism to divide findings into four different categories, presented here along with their definitions.

High

Risks classified as High are severe and immediate threats to your system's security and operations. These risks often have a high likelihood of occurrence and could lead to significant, potentially catastrophic, consequences affecting confidentiality, integrity, or availability. High risks require urgent attention and swift action to mitigate or remediate. They typically involve vulnerabilities that are easy to exploit and have a significant impact, such as those leading to data loss, financial damage, legal repercussions, or severe reputational harm.

Medium

Medium risks are significant concerns that pose a considerable threat to system security or data integrity but are less urgent than high risks. They might have a moderate likelihood of occurrence or impact, requiring attention and appropriate measures to mitigate. Medium risks might include vulnerabilities that are less likely to be exploited or have less severe impacts but still warrant timely intervention to prevent escalation or exploitation.

Low

Low risks are vulnerabilities or threats with minimal impact and likelihood of occurrence. They represent minor issues that do not significantly affect the system's overall security posture. While they require attention and should be addressed to maintain a robust security stance, they do not pose immediate or substantial threats.

Informational

Informational entries are not risks per se. They are included to suggest potential areas for improvement, or note deviations from best practices. These findings are typically items that do not have a direct or immediate impact on security but may influence security posture over time. They are valuable for comprehensive understanding, future planning, or awareness.

Glossary

Term	Description
EOA	Externally Owned Account
MPC	Multi-party computation
LZ	LayerZero