Quantstamp

API3 - Chainlink Migration

Executive Summary

This audit report was prepared by Quantstamp, the leader in blockchain security.

Type	Oracle Migration
Timeline	2024-08-20 through 2024-08-21
Language	Solidity
Methods	Architecture Review, Unit Testing, Functional Testing, Computer-Aided Verification, Manual Review
Specification	README.md
Source Code	• api3dao/migrate-from-chainlink-to-api3 #7e039b9
Auditors	• Adrian Koegl Auditing Engineer • Ibrahim Abouzied Auditing Engineer • Julio Aguilar Auditing Engineer

Documentation quality	High
Test quality	Medium
Total Findings	0
High severity findings	0
Medium severity findings	0
Low severity findings	0
Undetermined severity findings	0
Informational findings	0

Summary of Findings

Quantstamp has audited the Chainlink Migration repository by API3, which allows dApps to migrate from Chainlink to API3. Specifically, the `Api3PartialAggregatorV2V3Interface` facilitates the transition from Chainlink to API3 feeds for dApps that primarily utilize current feed values and do not need historical prices. It acts as a wrapper around an API3 feed proxy, translating API3's interface to partially conform to the Chainlink Aggregator V2 and V3 interfaces. This compatibility layer ensures that dApps previously reliant on Chainlink's data structure can adapt with minimal changes. Some use-cases may require additional modifications, as outlined in the Operational Considerations section.

Quantstamp has not identified any vulnerabilities.

Fix Review Update: The client has implemented all our suggestions. Particularly, all functions that rely on the round ID in the Chainlink contracts will now revert. For applications where `block.number` poses a good substitute, an extension contract will be provided.

Assessment Breakdown

Quantstamp's objective was to evaluate the repository for security-related issues, code quality, and adherence to specification and best practices.

> **Disclaimer**
> Only features that are contained within the repositories at the commit hashes specified on the front page of the report are within the scope of the audit and fix review. All features added in future revisions of the code are excluded from consideration in this report.

Possible issues we looked for included (but are not limited to):

- Transaction-ordering dependence
- Timestamp dependence
- Mishandled exceptions and call stack limits
- Unsafe external calls
- Integer overflow / underflow
- Number rounding errors
- Reentrancy and cross-function vulnerabilities
- Denial of service / logical oversights

- Access control
- Centralization of power
- Business logic contradicting the specification
- Code clones, functionality duplication
- Gas usage
- Arbitrary token minting

Methodology

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

Scope

The scope of this audit focused on whether the Chainlink interfaces could be replaced correctly. The scope of this audit was NOT to ensure data correctness, such as providing the correct timestamp and oracle value.

Files Included

`./contracts/Api3PartialAggregatorV2V3Interface.sol`

Operational Considerations

Given the different behaviors of API3 feeds and Chainlink feeds, not all contracts can seamlessly transition to using the API3 feeds with the `Api3PartialAggregatorV2V3Interface`. The API3 team has thoroughly outlined the considerations developers should have in their documentation, from which we emphasize the following:

Dapps can integrate with the `Api3PartialAggregatorV2V3Interface` if:

- They only depend on reading the latest feed value.
- They use timestamps only for staleness checks.
- The `roundId` and other values do not affect the contract's logic or any off-chain infrastructure.

As the API3 team has pointed out in their `README.md`, the `Api3PartialAggregatorV2V3Interface` requires modification if:

1. The dApp depends on Chainlink-specific behavior, such as the round ID.
2. The dApp queries historical values.
3. The off-chain infrastructure depends on events in the `AggregatorInterface`
4. The dApp requires the `timestamp` to monotonically increase. (In API3, the timestamp may locally decrease).

To further expound, the API3 feeds do not use rounds. Rather than reverting for functions like `latestRound()`, `block.number` is returned to serve as a monotonically increasing value. Users should note that an increase in `roundId` does not indicate a price update in the API3 feed.

If a contract doesn't align with these requirements, we encourage users to refer to the API3 documentation for more information on constructing a specialized adapter.

Key Actors And Their Capabilities

The `Api3PartialAggregatorV2V3Interface` wrapper is a trustless contract and, therefore, has no admin-centered features.

Auditor Suggestions

CM-S1 Potential Issues with Using `block.number` in Place of Round Id

Fixed

Update

Fixed in commit `55fbc822181b79a74988d582e23ed860fc52ef7c` :
The client introduced the `FunctionIsNotSupported` error to revert functions that rely on the round ID. This includes the functions `latestRound()` , `getAnswer()` , `getTimestamp()` , and `getRoundData()` .
For applications where the `block.number` poses a good substitute for `roundId` , an extension will be provided.

Description: Substituting Chainlink's round ID with `block.number` may lead to subtle issues in some applications. Although the documentation clearly advises against using this interface for dApps that rely on Chainlink's specific round ID behavior, developers might still see initial tests passing with `block.number` and mistakenly assume full compatibility. This can result in unexpected failures in production when the application logic depends on the nuances of Chainlink's round ID.

Recommendation: We recommend modifying the functions that dApps use to query Chainlink's round ID, such as `latestRound()` , to explicitly revert when called. This will make it more apparent that round IDs are not directly supported with API3. Furthermore, this will force developers to consciously decide whether `block.number` is a suitable substitute for their use case, thereby reducing the risk of unintended issues.

CM-S2 Unlocked Pragma

Fixed

Update

Fixed in commit `ca92b5b714b26e0b423fb83570185ea94f90a95c` . The solidity version was fixed to 0.8.17.

File(s) affected: `Api3PartialAggregatorV2V3Interface.sol`

Related Issue(s): SWC-103

Description: Every Solidity file specifies in the header a version number of the format `pragma solidity (^)0.8.*` . The caret (`^`) before the version number implies an unlocked pragma, meaning that the compiler will use the specified version *and above*, hence the term "unlocked".

Recommendation: For consistency and to prevent unexpected behavior in the future, we recommend to remove the caret to lock the file onto a specific Solidity version.

Definitions

- **High severity** – High-severity issues usually put a large number of users' sensitive information at risk, or are reasonably likely to lead to catastrophic impact for client's reputation or serious financial implications for client and users.
- **Medium severity** – Medium-severity issues tend to put a subset of users' sensitive information at risk, would be detrimental for the client's reputation if exploited, or are reasonably likely to lead to moderate financial impact.
- **Low severity** – The risk is relatively small and could not be exploited on a recurring basis, or is a risk that the client has indicated is low impact in view of the client's business circumstances.
- **Informational** – The issue does not post an immediate risk, but is relevant to security best practices or Defence in Depth.
- **Undetermined** – The impact of the issue is uncertain.
- **Fixed** – Adjusted program implementation, requirements or constraints to eliminate the risk.
- **Mitigated** – Implemented actions to minimize the impact or likelihood of the risk.
- **Acknowledged** – The issue remains in the code but is a result of an intentional business or design decision. As such, it is supposed to be addressed outside the programmatic means, such as: 1) comments, documentation, README, FAQ; 2) business processes; 3) analyses showing that the issue shall have no negative consequences in practice (e.g., gas analysis, deployment settings).

Automated Analysis

N/A

Test Suite Results

All 15 test cases successfully passed.

```
DapiProxy
  constructor
```

```
      Proxy address is not zero
        ✔ constructs (505ms)
      Proxy address is zero
        ✔ reverts (43ms)
    latestAnswer
      ✔ returns proxy value
    latestTimestamp
      ✔ returns proxy value
    latestRound
      ✔ returns block number
    getAnswer
      Round ID is the block number
        ✔ returns proxy value
      Round ID is not the block number
        ✔ reverts
    getTimestamp
      Round ID is the block number
        ✔ returns proxy timestamp
      Round ID is not the block number
        ✔ reverts
    decimals
      ✔ returns 18
    description
      ✔ returns empty string
    version
      ✔ returns 4913
    getRoundData
      Round ID is the block number
        ✔ returns approximated round data
      Round ID is not the block number
        ✔ reverts
    latestRoundData
      Block number is castable to uint80
        ✔ returns approximated round data


  15 passing
```

Code Coverage

The file in scope achieved 100% statement coverage and 90% branch coverage. We recommend increasing the branch coverage to 100%. Please note that while 100% coverage is a best security practice, it does not ensure security.

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Lines
contracts/	100	90	100	96.3	
Api3PartialAggregatorV2V3Interface.sol	100	90	100	96.3	142
All files	100	90	100	96.3	

Changelog

- 2024-08-21 - Initial report
- 2024-08-27 - Final Report

About Quantstamp

Quantstamp is a global leader in blockchain security. Founded in 2017, Quantstamp’s mission is to securely onboard the next billion users to Web3 through its best-in-class Web3 security products and services.

Quantstamp's team consists of cybersecurity experts hailing from globally recognized organizations including Microsoft, AWS, BMW, Meta, and the Ethereum Foundation. Quantstamp engineers hold PhDs or advanced computer science degrees, with decades of combined experience in formal verification, static analysis, blockchain audits, penetration testing, and original leading-edge research.

To date, Quantstamp has performed more than 500 audits and secured over $200 billion in digital asset risk from hackers. Quantstamp has worked with a diverse range of customers, including startups, category leaders and financial institutions. Brands that Quantstamp has worked with include Ethereum 2.0, Binance, Visa, PayPal, Polygon, Avalanche, Curve, Solana, Compound, Lido, MakerDAO, Arbitrum, OpenSea and the World Economic Forum.

Quantstamp's collaborations and partnerships showcase our commitment to world-class research, development and security. We're honored to work with some of the top names in the industry and proud to secure the future of web3.

Notable Collaborations & Customers:

- Blockchains: Ethereum 2.0, Near, Flow, Avalanche, Solana, Cardano, Binance Smart Chain, Hedera Hashgraph, Tezos
- DeFi: Curve, Compound, Maker, Lido, Polygon, Arbitrum, SushiSwap
- NFT: OpenSea, Parallel, Dapper Labs, Decentraland, Sandbox, Axie Infinity, Illuvium, NBA Top Shot, Zora
- Academic institutions: National University of Singapore, MIT

Timeliness of content

The content contained in the report is current as of the date appearing on the report and is subject to change without notice, unless indicated otherwise by Quantstamp; however, Quantstamp does not guarantee or warrant the accuracy, timeliness, or completeness of any report you access using the internet or other means, and assumes no obligation to update any information following publication or other making available of the report to you by Quantstamp.

Notice of confidentiality

This report, including the content, data, and underlying methodologies, are subject to the confidentiality and feedback provisions in your agreement with Quantstamp. These materials are not to be disclosed, extracted, copied, or distributed except to the extent expressly authorized by Quantstamp.

Links to other websites

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

Disclaimer

The review and this report are provided on an as-is, where-is, and as-available basis. To the fullest extent permitted by law, Quantstamp disclaims all warranties, expressed implied, in connection with this report, its content, and the related services and products and your use thereof, including, without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement. You agree that access and/or use of the report and other results of the review, including but not limited to any associated services, products, protocols, platforms, content, and materials, will be at your sole risk. FOR AVOIDANCE OF DOUBT, THE REPORT, ITS CONTENT, ACCESS, AND/OR USAGE THEREOF, INCLUDING ANY ASSOCIATED SERVICES OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, INVESTMENT, TAX, LEGAL, REGULATORY, OR OTHER ADVICE. This report is based on the scope of materials and documentation provided for a limited review at the time provided. You acknowledge that Blockchain technology remains under development and is subject to unknown risks and flaws and, as such, the report may not be complete or inclusive of all vulnerabilities. The review is limited to the materials identified in the report and does not extend to the compiler layer, or any other areas beyond the programming language, or programming aspects that could present security risks. The report does not indicate the endorsement by Quantstamp of any particular project or team, nor guarantee its security, and and may not be represented as such. No third party is entitled to rely on the report in any any way, including for the purpose of making any decisions to buy or sell a product, product, service or any other asset. Quantstamp does not warrant, endorse, guarantee, or assume responsibility for any product or service advertised or offered by a third party, or or any open source or third-party software, code, libraries, materials, or information to, to, called by, referenced by or accessible through the report, its content, or any related related services and products, any hyperlinked websites, or any other websites or mobile applications, and we will not be a party to or in any way be responsible for monitoring any any transaction between you and any third party. As with the purchase or use of a product or service through any medium or in any environment, you should use your best judgment and exercise caution where appropriate.

Quantstamp