Bao.Finance process quality review

Score: 38%

Overview

This is a Process Quality Review of bao.finance completed on May 10, 2021. It was performed using the Process Review process (version 0.7) and is documented here. The review was performed by Lucas of DeFiSafety. Check out our Telegram.

The final score of the review is 38%, a fail. The breakdown of the scoring is in Scoring Appendix. For our purposes, a pass is 70%.

Summary of the Process

Very simply, the review looks for the following declarations from the developer's site. With these declarations, it is reasonable to trust the smart contracts.

- **Here are my smart contracts on the blockchain**
- **Here is the documentation that explains what my smart contracts do**
- **Here are the tests I ran to verify my smart contract**
- **Here are the audit(s) performed on my code by third party experts**
- **Here are the admin controls and strategies**

Disclaimer

This report is for informational purposes only and does not constitute investment advice of any kind, nor does it constitute an offer to provide investment advisory or other services. Nothing in this report shall be considered a solicitation or offer to buy or sell any security, token, future, option or other financial instrument or to offer or provide any investment advice or service to any person in any jurisdiction. Nothing contained in this report constitutes investment advice or offers any opinion with respect to the suitability of any security, and the views expressed in this report should not be taken as advice to buy, sell or hold any security. The information in this report should not be relied upon for the purpose of investing. In preparing the information contained in this report, we have not taken into account the investment needs, objectives and

financial circumstances of any particular investor. This information has no regard to the specific investment objectives, financial situation and particular needs of any specific recipient of this information and investments discussed may not be suitable for all investors.

Any views expressed in this report by us were prepared based upon the information available to us at the time such views were written. The views expressed within this report are limited to DeFiSafety and the author and do not reflect those of any additional or third party and are strictly based upon DeFiSafety, its authors, interpretations and evaluation of relevant data. Changed or additional information could cause such views to change. All information is subject to possible correction. Information may quickly become unreliable for various reasons, including changes in market conditions or economic circumstances.

Chain

This section indicates the blockchain used by this protocol.

> **Chain: Binance**

Guidance:
Ethereum
Binance

Code and Team

This section looks at the code deployed on the Mainnet that gets reviewed and its corresponding software repository. The document explaining these questions is here. This review will answer the questions;

1) Are the executing code addresses readily available? (%)
2) Is the code actively being used? (%)
3) Is there a public software repository? (Y/N)

4) Is there a development history visible? (%)
5) Is the team public (not anonymous)? (Y/N)

1) Are the executing code addresses readily available? (%)

> Answer: 100%

Guidance:

100% Clearly labelled and on website, docs or repo, quick to find
70% Clearly labelled and on website, docs or repo but takes a bit of looking
40% Addresses in mainnet.json, in discord or sub graph, etc
20% Address found but labelling not clear or easy to find
0% Executing addresses could not be found

They are available at website https://docs.bao.finance/contracts-and-key-info/ethereum-main-net as indicated in the Appendix.

How to improve this score

Make the Ethereum addresses of the smart contract utilized by your application available on either your website or your GitHub (in the README for instance). Ensure the addresses is up to date. This is a very important question wrt to the final score.

2) Is the code actively being used? (%)

> Answer: 100%

Activity is 24 transactions a day on contract *BaoMasterFarmer.sol*, as indicated in the Appendix.

Percentage Score Guidance

100% More than 10 transactions a day
70% More than 10 transactions a week
40% More than 10 transactions a month

10% Less than 10 transactions a month
0% No activity

3) Is there a public software repository? (Y/N)

⚠ Answer: No

There is a BaoSwap repository but not a Bao Finance repository, containing the relevent solidity contracts.

Is there a public software repository with the code at a minimum, but normally test and scripts also (Y/N). Even if the repo was created just to hold the files and has just 1 transaction, it gets a Yes. For teams with private repos, this answer is No.

4) Is there a development history visible? (%)

⚠ Answer: 0%

As a private repository there is no development history visible.

This checks if the software repository demonstrates a strong steady history. This is normally demonstrated by commits, branches and releases in a software repository. A healthy history demonstrates a history of more than a month (at a minimum).

Guidance:
100% Any one of 100+ commits, 10+branches
70% Any one of 70+ commits, 7+branches
50% Any one of 50+ commits, 5+branches
30% Any one of 30+ commits, 3+branches
0% Less than 2 branches or less than 10 commits

How to improve this score

Continue to test and perform other verification activities after deployment, including routine maintenance updating to new releases of testing and deployment tools. A public development

history indicates clearly to the public the level of continued investment and activity by the developers on the application. This gives a level of security and faith in the application.

5) Is the team public (not anonymous)? (Y/N)

⚠ Answer: No

The Bao Finance team is not public. Their creator is active on twitter, but he has not revealed his name.

For a yes in this question the real names of some team members must be public on the website or other documentation. If the team is anonymous and then this question is a No.

Documentation

This section looks at the software documentation. The document explaining these questions is here.

Required questions are;

6) Is there a whitepaper? (Y/N)
7) Are the basic software functions documented? (Y/N)
8) Does the software function documentation fully (100%) cover the deployed contracts? (%)
9) Are there sufficiently detailed comments for all functions within the deployed contract code (%)
10) Is it possible to trace from software documentation to the implementation in code (%)

6) Is there a whitepaper? (Y/N)

✓ Answer: Yes

Location: https://docs.bao.finance/

How to improve this score

Ensure the white paper is available for download from your website or at least the software repository. Ideally update the whitepaper to meet the capabilities of your present application.

7) Are the basic software functions documented? (Y/N)

> Answer: Yes

The functions that the contract owner has control over can be found in their documentation.

How to improve this score

Write the document based on the deployed code. For guidance, refer to the SecurEth System Description Document.

8) Does the software function documentation fully (100%) cover the deployed contracts? (%)

> Answer: 50%

They document a few functions in the "Contract changes" section of their documentation.

Guidance:

100% All contracts and functions documented
80% Only the major functions documented
79-1% Estimate of the level of software documentation
0% No software documentation

How to improve this score

This score can improve by adding content to the requirements document such that it comprehensively covers the requirements. For guidance, refer to the SecurEth System Description Document . Using tools that aid traceability detection will help.

9) Are there sufficiently detailed comments for all functions within the deployed contract code (%)

> Answer: 73%

This comment/code ratio was tested based on the contracts published on etherscan. the CtC ratio is 73%.

The Comments to Code (CtC) ratio is the primary metric for this score.

Guidance:
100% CtC > 100 Useful comments consistently on all code
90-70% CtC > 70 Useful comment on most code
60-20% CtC > 20 Some useful commenting
0% CtC < 20 No useful commenting

How to improve this score

This score can improve by adding comments to the deployed code such that it comprehensively covers the code. For guidance, refer to the SecurEth Software Requirements.

10) Is it possible to trace from software documentation to the implementation in code (%)

> Answer: 60%

There is clear traceability for the functions that are described in the documentation, but not all the functions.

Guidance:

100% Clear explicit traceability between code and documentation at a requirement level for all code

60% Clear association between code and documents via non explicit traceability

40% Documentation lists all the functions and describes their functions

0% No connection between documentation and code

How to improve this score

This score can improve by adding traceability from requirements to code such that it is clear where each requirement is coded. For reference, check the SecurEth guidelines on traceability.

Testing

This section looks at the software testing available. It is explained in this document. This section answers the following questions;

11) Full test suite (Covers all the deployed code) (%)
12) Code coverage (Covers all the deployed lines of code, or explains misses) (%)
13) Scripts and instructions to run the tests (Y/N)
14) Report of the results (%)
15) Formal Verification test done (%)
16) Stress Testing environment (%)

11) Is there a Full test suite? (%)

> Answer: 40%

With a Test to Code ratio of 28%, this is not a very robust test suite.

This score is guided by the Test to Code ratio (TtC). Generally a good test to code ratio is over 100%. However the reviewers best judgement is the final deciding factor.

Guidance:

100% TtC > 120% Both unit and system test visible

80% TtC > 80% Both unit and system test visible
40% TtC < 80% Some tests visible
0% No tests obvious

How to improve this score

This score can improve by adding tests to fully cover the code. Document what is covered by traceability or test results in the software repository.

12) Code coverage (Covers all the deployed lines of code, or explains misses) (%)

Answer: 30%

There is no evident report of code coverage.

Guidance:
100% Documented full coverage
99-51% Value of test coverage from documented results
50% No indication of code coverage but clearly there is a reasonably complete set of tests
30% Some tests evident but not complete
0% No test for coverage seen

How to improve this score

This score can improve by adding tests achieving full code coverage. A clear report and scripts in the software repository will guarantee a high score.

13) Scripts and instructions to run the tests (Y/N)

Answer: No

There are no tests to run or instructions on how to test.

How to improve this score

Add the scripts to the repository and ensure they work. Ask an outsider to create the environment and run the tests. Improve the scripts and docs based on their feedback.

14) Report of the results (%)

Answer: 0%

There is no evident report of the test results.

Guidance:
100% Detailed test report as described below
70% GitHub Code coverage report visible
0% No test report evident

How to improve this score

Add a report with the results. The test scripts should generate the report or elements of it.

15) Formal Verification test done (%)

Answer: 0%

There is no evidence of any formal verification.

16) Stress Testing environment (%)

Answer: 0%

there are no evident Kovan or Ropsten testnet addresses.

Security

This section looks at the 3rd party software audits done. It is explained in this document. This section answers the following questions;

17) Did 3rd Party audits take place? (%)
18) Is the bounty value acceptably high?

17) Did 3rd Party audits take place? (%)

⚠ Answer: 20%

The developer has stated that no audit has been completed.

Guidance:
100% Multiple Audits performed before deployment and results public and implemented or not required
90% Single audit performed before deployment and results public and implemented or not required
70% Audit(s) performed after deployment and no changes required. Audit report is public
20% No audit performed
0% Audit Performed after deployment, existence is public, report is not public and no improvements deployed OR smart contract address' not found, question

18) Is the bounty value acceptably high (%)

⚠ Answer: 0%

There is no bug bounty program offered.

100% Bounty is 10% TVL or at least $1M AND active program (see below)
90% Bounty is 5% TVL or at least 500k AND active program
80% Bounty is 5% TVL or at least 500k

70% Bounty is 100k or over AND active program
50% Bounty is 100k or over
40% Bounty is 50k or over
20% Bug bounty program bounty is less than 50k
0% No bug bounty program offered

Active program means a third party actively driving hackers to the site. Inactive program would be static mention on the docs.

Access Controls

This section covers the documentation of special access controls for a DeFi protocol. The admin access controls are the contracts that allow updating contracts or coefficients in the protocol. Since these contracts can allow the protocol admins to "change the rules", complete disclosure of capabilities is vital for user's transparency. It is explained in this document. The questions this section asks are as follow;

19) Can a user clearly and quickly find the status of the admin controls?
20) Is the information clear and complete?
2`) Is the information in non-technical terms that pertain to the investments?
22) Is there Pause Control documentation including records of tests?

19) Can a user clearly and quickly find the status of the admin controls (%)

> ⓘ Answer: 70%

Location: https://docs.bao.finance/contract-changes

Guidance:
100% Clearly labelled and on website, docs or repo, quick to find
70% Clearly labelled and on website, docs or repo but takes a bit of looking
40% Access control docs in multiple places and not well labelled
20% Access control docs in multiple places and not labelled
0% Admin Control information could not be found

20) Is the information clear and complete (%)

Answer: 60%

The type of ownership is clearly indicated (OnlyOwner / MultiSig / Defined Roles) 30% AND
The capabilities for change in the contracts are described 30%

Guidance:
All the contracts are immutable -- 100% OR

All contracts are clearly labelled as upgradeable (or not) -- 30% AND
The type of ownership is clearly indicated (OnlyOwner / MultiSig / Defined Roles) -- 30% AND
The capabilities for change in the contracts are described -- 30%

How to improve this score

Create a document that covers the items described above. An example is enclosed.

21) Is the information in non-technical terms that pertain to the investments (%)

Answer: 90%

The admin controls are well-described in investor-friendly language in their "contract changes" documentation.

Guidance:

100%	All the contracts are immutable
90%	Description relates to investments safety and updates in clear, complete non-software l language
30%	Description all in software specific language
0%	No admin control information could not be found

How to improve this score

Create a document that covers the items described above in plain language that investors can understand. An example is enclosed.

22) Is there Pause Control documentation including records of tests (%)

> ⚠ Answer: 0%

There is no evident pause control documentation.

Guidance:

100% All the contracts are immutable or no pause control needed and this is explained OR

100% Pause control(s) are clearly documented and there is records of at least one test within 3 months

80% Pause control(s) explained clearly but no evidence of regular tests

40% Pause controls mentioned with no detail on capability or tests

0% Pause control not documented or explained

How to improve this score

Create a document that covers the items described above in plain language that investors can understand. An example is enclosed.

Appendices

Author Details

The author of this review is Rex of DeFi Safety.

Email : rex@defisafety.com Twitter : @defisafety

I started with Ethereum just before the DAO and that was a wonderful education. It showed the importance of code quality. The second Parity hack also showed the importance of good process. Here my aviation background offers some value. Aerospace knows how to make reliable code using quality processes.

I was coaxed to go to EthDenver 2018 and there I started SecuEth.org with Bryant and Roman. We created guidelines on good processes for blockchain code development. We got EthFoundation funding to assist in their development.

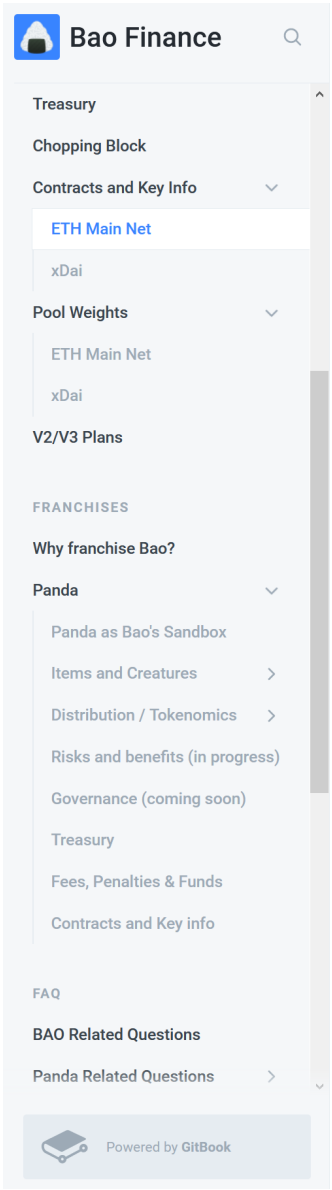
Process Quality Reviews are an extension of the SecurEth guidelines that will further increase the quality processes in Solidity and Vyper development.

DeFiSafety is my full time gig and we are working on funding vehicles for a permanent staff.

Scoring Appendix

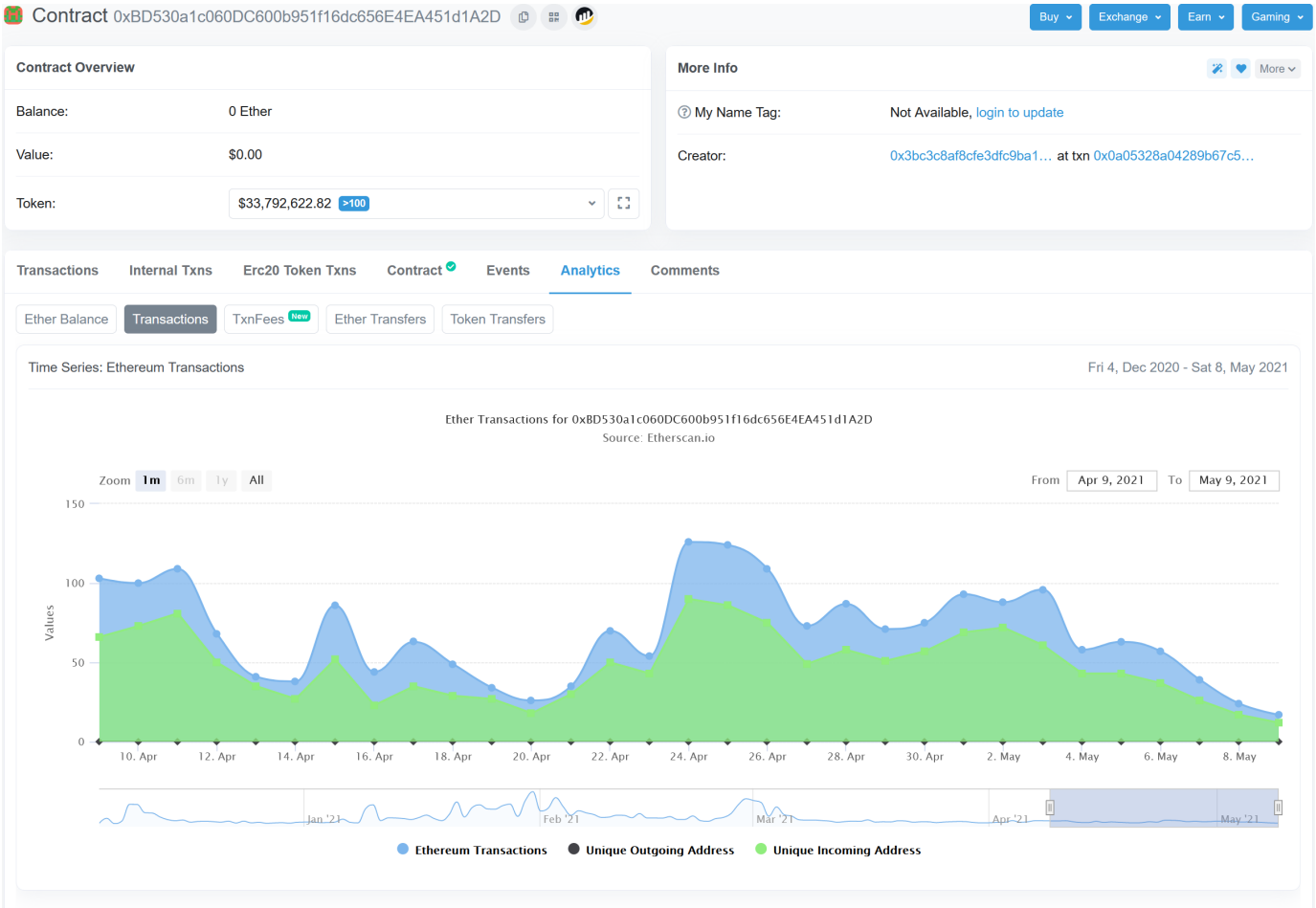
	Total	bao.finance	
PQ Audit Scoring Matrix (v0.7)	**Points**	**Answer**	**Points**
Total	**260**		99.25
Code and Team			**38%**
1) Are the executing code addresses readily available? (%)	**20**	100%	20
2) Is the code actively being used? (%)	**5**	100%	5
3) Is there a public software repository? (Y/N)	**5**	N	0
4) Is there a development history visible? (%)	**5**	0%	0
5) Is the team public (not anonymous)? (Y/N)	**15**	N	0
Code Documentation			
6) Is there a whitepaper? (Y/N)	**5**	y	5
7) Are the basic software functions documented? (Y/N)	**10**	Y	10
8) Does the software function documentation fully (100%) cover the deployed contracts? (%)	**15**	50%	7.5
9) Are there sufficiently detailed comments for all functions within the deployed contract code (%)	**5**	75%	3.75
10) Is it possible to trace from software documentation to the implementation in code (%)	**10**	60%	6
Testing			
11) Full test suite (Covers all the deployed code) (%)	**20**	40%	8
12) Code coverage (Covers all the deployed lines of code, or explains misses) (%)	**5**	30%	1.5
13) Scripts and instructions to run the tests? (Y/N)	**5**	0	0
14) Report of the results (%)	**10**	0%	0
15) Formal Verification test done (%)	**5**	0%	0
16) Stress Testing environment (%)	**5**	0%	0
Security			
17) Did 3rd Party audits take place? (%)	**70**	20%	14
18) Is the bug bounty acceptable high? (%)	**10**	0%	0
Access Controls			
19) Can a user clearly and quickly find the status of the admin controls	**5**	70%	3.5
20) Is the information clear and complete	**10**	60%	6
21) Is the information in non-technical terms	**10**	90%	9
22) Is there Pause Control documentation including records of tests	**10**	0%	0
Section Scoring			
Code and Team	50	50%	
Documentation	45	72%	
Testing	50	19%	
Security	80	18%	
Access Controls	35	53%	

Executing Code Appendix



BAO	0x374cb8c27130e2c9e04f44303f3c8351b9de61c1	0x9973bb0fe5f8df5de730776df09e
wBTC	0x2260fac5e5542a773aa44fbcfedf7c193bc2c599	0xbb2b8038a1640196fbe3e38816f3
Tether	0xdac17f958d2ee523a2206206994597c13d831ec7	0x0d4a11d5eeaac28ec3f61d100daf
ChainLink	0x514910771af9ca656af840dff83e8264ecf986ca	0xa2107fa5b38d9bbd2c461d6edf11
USDC	0xa0b86991c6218b36c1d19d4a2e9eb0ce3606eb48	0xb4e16d0168e52d35cacd2c6185b4
cDAI	0x5d3a536e4d6dbd6114cc1ead35777bab948e3643	0x9896bd979f9da57857322cc15e15
OKB	0x75231f58b43240c9718dd58b4967c5114342a86c	0x17782d58c715aa2a4458d5fb1c1c
LEO	0x2af5d2ad76741191d15dfe7bf6ac92d4bd912ca3	0x523a36ad73c402e456f49b04f0fe
DAI	0x6b175474e89094c44da98b954eedeac495271d0f	0xa478c2975ab1ea89e8196811f51a
UNI	0x1f9840a85d5af5bf1d1762f925bdaddc4201f984	0xd3d2e2692501a5c9ca623199d388
HT	0x6f259637dcd74c767781e37bc6133cd6a68aa161	0x26ce49c08ee71aff0c43db8f8b9b
AAVE	0x7fc66500c84a76ad7e9c93437bfc5ac33e2ddae9	0xdfc14d2af169b0d36c4eff567ada
CEL	0xaaaebe6fe48e54f431b0c390cfaf0b017d09d42d	0xa5e79baee540f000ef6f23d067cd
SNX	0xc011a73ee8576fb46f5e1c5751ca3b9fe0af2a6f	0x43ae24960e5534731fc831386c07
CRV	0xd533a949740bb3306d119cc777fa900ba034cd52	0x3da1313ae46132a397d90d95b142
YFI	0x0bc529c00c6401aef6d220be8c6ea1667f6ad93e	0x2fdbadf3c4d5a8666bc06645b835
COMP	0xc00e94cb662c3520282e6f5717214004a7f26888	0xcffdded873554f362ac02f8fb1f02
MKR	0x9f8f72aa9304c8b593d555f12ef6589cc3a579a2	0xc2adda861f89bbb333c90c492cb8
UMA	0x04fa0d235c4abf4bcf4787af4cf447de572ef828	0x88d97d199b9ed37c29d846d00d44
FTT	0x50d1c9771902476076ecfc8b2a83ad6b9355a4c9	0xf04543fbf20daee9b0357db96642
RENBTC	0xeb4c2781e4eba804ce9a9803c67d0893436bb27d	0x81fbef4704776cc5bba0a5df3a90
BAT	0x0d8775f648430679a709e98d2b0cb6250d2887ef	0xb6909b960dbbe7392d405429eb2b
TUSD	0x0000000000085d4780b73119b644ae5ecd22b376	0xb4d0d9df2738abe81b87b66c8085
HUSD	0xdf574c24545e5ffecb9a659c229253d4111d87e1	0x8749068c5b45fdaa369319e5daa1

Code Used Appendix



SLOC Appendix

Solidity Contracts

Language	Files	Lines	Blanks	Comments	Code	Complexity
Solidity	1	4610	669	1675	2266	284

Comments to Code 1675/2266 = 73%

Javascript Tests

Language	Files	Lines	Blanks	Comments	Code	Complexity
JavaScript	3	1119	132	335	652	25

Tests to Code 652/2266 = 28%