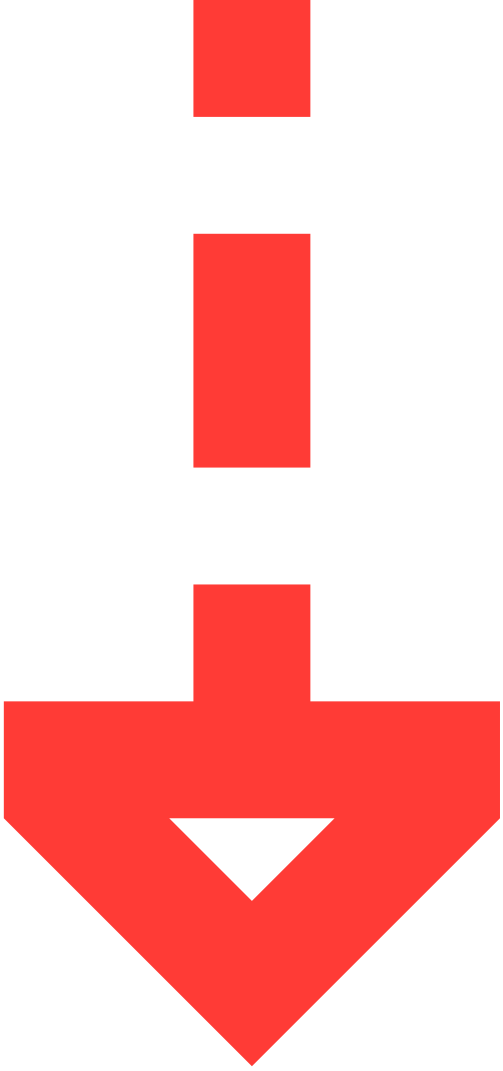
clarity.

Contract Audit



REPORT DATE	**November 9th, 2018**
REPORT VERSION	**2.0**

PREPARED BY

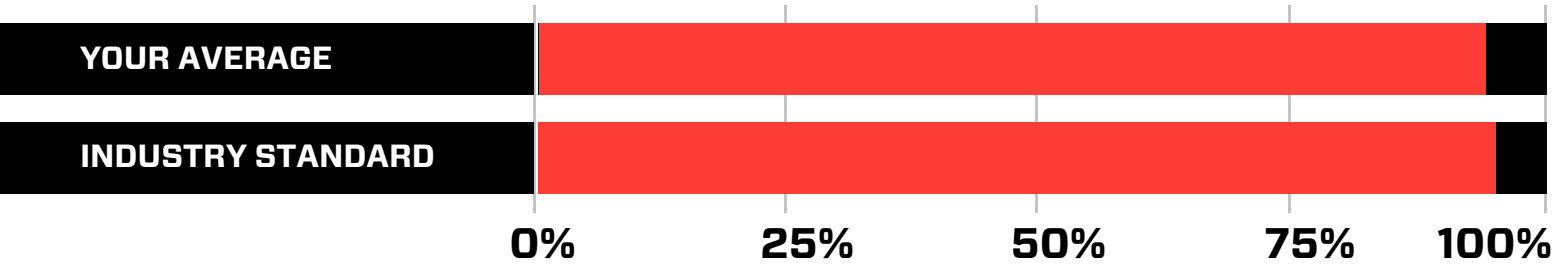
HOSHO

This document outlines the overall security of Clarity Project's smart contract as evaluated by Hosho's Smart Contract auditing team. The scope of this audit was to analyze and document Clarity Project's token contract codebase for quality, security, and correctness.

Contract Status

No issues were discovered in this contract during the auditing process. (See Complete Analysis)

Testable Code



Testable code is 93.81%, which is on par with the industry standard of 95%. (See Coverage Report)

It should be noted that this audit is not an endorsement of the reliability or effectiveness of the contract, rather limited to an assessment of the logic and implementation. In order to ensure a secure contract that's able to withstand the Ethereum network's fast-paced and rapidly changing environment, we at Hosho recommend that the Clarity Project team put in place a bug bounty program to encourage further and active analysis of the smart contract.

Table of Contents

Auditing Strategy and Techniques Applied

The Hosho team has performed a thorough review of the smart contract code, the latest version as written and updated on October 30th, 2018. All main contract files were reviewed using the following tools and processes. (See All Files Covered)

Throughout the review process, care was taken to ensure that the token contract:

- Implements and adheres to existing ERC-20 Token standards appropriately and effectively;
- Documentation and code comments match logic and behavior;
- Distributes tokens in a manner that matches calculations;
- Follows best practices in efficient use of gas, without unnecessary waste;
- Uses methods safe from reentrance attacks; and
- Is not affected by the latest vulnerabilities.

The Hosho team has followed best practices and industry-standard techniques to verify the implementation of Clarity Project's token contract. To do so, the code is reviewed line-by-line by our team of expert pentesters and smart contract developers, documenting any issues as they are discovered. Part of this work includes writing a unit test suite using the Meadow testing framework. In summary, our strategies consist largely of manual collaboration between multiple team members at each stage of the review:

1	**Due diligence in assessing the overall code quality of the codebase.**
2	**Cross-comparison with other, similar smart contracts by industry leaders.**
3	**Testing contract logic against common and uncommon attack vectors.**
4	**Thorough, manual review of the codebase, line-by-line.**
5	**Deploying the smart contract to testnet and production networks using multiple client. implementations to run live tests.**

2.1 Summary

The Clarity Project contracts form a standard ERC-20 Token, as well as a timed crowdsale event. The token contract has additional mintable, burnable, and multiple ownable components included that add expanded functionality to the base token. The crowdsale functionality makes use of timed, post-deliverable, and finalizable contracts to create a two-phase purchase system with a bonus token payout included in the referral system.

2.2 Coverage Report

As part of our work assisting Clarity Project in verifying the correctness of their contract code, our team was responsible for writing a unit test suite using the Meadow testing framework.

- Branches: 91.96%
- Functions: 92.94%
- Lines: 96.54%

2.3 Failing Tests

No failing tests!

For ease of navigation, sections are arranged from most critical to least critical. Issues are tagged “Resolved” or “Unresolved” depending on whether they have been fixed or still need addressing. Furthermore, the severity of each issue is written as assessed by the risk of exploitation or other unexpected or otherwise unsafe behavior:

Critical

The issue affects the contract in such a way that funds may be lost, allocated incorrectly, or otherwise result in a significant loss.

High

The issue affects the ability of the contract to compile or operate in a significant way.

Medium

The issue affects the ability of the contract to compile or operate in a significant way.

Low

The issue has minimal impact on the contract’s ability to operate.

Informational

The issue has no impact on the contract’s ability to operate, and is meant only as additional information.

3.1 Resolved: Unused Contract

LOW

Contract: ERC20

Explanation

ClarityCrowdsale imports the entire ERC20 contract but does not use it in any portion of the crowdsale functionality. SafeERC20 is used during token transfers which renders the transfer function from the ERC20 contract useless.

Resolution

The Clarity team has removed the ERC20 contract from the ClarityCrowdsale file.

We are grateful to have been given the opportunity to work with the Clarity Project team.

The Clarity Project contracts create a robust ERC-20 token contract and a two-phase crowdsale with bonus system. All previously found issues have been addressed by the Clarity team, and these contracts have now passed the rigorous auditing process performed by the Hosho team.

The statements made in this document should not be interpreted as investment or legal advice, nor should its authors be held accountable for decisions made based on them.

We at Hosho recommend that the Clarity Project team put in place a bug bounty program to encourage further analysis of the smart contract by other third parties.

Test Suite Results

Contract: HoshoAudit.BurnableTests
- √ burnFrom_NoAllowance_ExpectRevert (0.3579600s)
- √ burnFrom_FromAccount_EmitTransferEvent (0.4803970s)
- √ burn_FromAccount_EmitTransferEvent (0.4847440s)
- √ mint_MintToAddressZero_ExpectRevert (0.3373940s)
- √ burn_BurnMoreThanBalance_ExpectRevert (0.3580200s)
- √ burn_AddressisZero_ExpectRevert (0.3503060s)

Contract: HoshoAudit.CrowdsaleOwnableTests
- √ transferOwnership_calledByOwner_UpdateOwnerVariable (0.0626890s)
- √ transferOwnership_calledByOwner_EmitOwnershipTransferredEvent (0.0516970s)
- √ transferOwnership_newOwnerIsZeroAddress_ExpectRevert (0.0200050s)
- √ renounceOwnership_calledByOwner_EmitOwnershipTransferredEvent (0.1545570s)
- √ transferOwnership_senderIsNonOwner_ExpectRevert (0.0959670s)

Contract: HoshoAudit.CrowdsaleTests
- √ transferFrom_valueGreatThanAllowed_ExpectRevert (0.1583610s)
- √ buyTokens_SaleClosed_ExpectRevert (0.0864140s)
- √ transfer_ToAddressZero_ExpectRevert (0.0189380s)
- √ transfer_SendMoreThanBalance_ExpectRevert (0.0885650s)
- √ finalized_SaleNotFinalized_ReturnFalse (0.0311180s)
- √ indexed - Event Transfer defines parameter 0 at index from of type address and indexed = True (0.7792390s)
- √ removeAdmin_CalledByAdmin_AssertIsAdmin (0.1061440s)
- √ finalize_SenderIsNotFounder_ExpectRevert (0.1254100s)
- √ withdrawTokens_AmountIsZero_ExpectRevert (0.2187770s)
- √ buyTokens_AccountNotWhitelisted_ExpectRevert (0.0427170s)
- √ withdrawTokens_SaleNotClosed_ExpectRevert (0.0401970s)
- √ buyTokens_SaleEndsWithTokensLeft_TransferTokensLeftToWallet (0.8186600s)
- √ decreaseApproval_Success (0.1525870s)
- √ buyTokens_SaleOpen_EmitTokensPurchased (0.7414810s)
- √ buyTokens_BeneficiaryIsAddressZero_ExpectRevert (0.1673980s)
- √ transferFrom_ToAccountZero_ExpectRevert (0.0934620s)
- √ rate_RateAmount_AssertRate (0.2087780s)
- √ balanceOf_CheckOwnerBalance_AssertEqual (0.0998370s)

Appendix A

Contract: HoshoAudit.CrowdsaleTests
- √ wallet_WalletAddress_ReturnWallet (0.3716080s)
- √ opentingTime_CheckOpening_AssertTime (0.3645850s)
- √ transferFrom_ApproveThenTransfer_EmitEvent (0.2012430s)
- √ finalize_AlreadyFinalized_ExpectRevert (0.2270730s)
- √ addAdmin_CalledByAdmin_UpdateAdmins (0.0888990s)
- √ crowdsaleConstructor_TokenIsAddressZero_ExpectRevert (0.0191760s)

Contract: HoshoAudit.CrowdsaleTests
- √ transferFrom_valueGreaterThanBalance_ExpectRevert (0.0491020s)
- √ buyTokens_BuyZeroWei_ExpectRevert (0.1676510s)
- √ closingTime_CheckClosing_AssertTime (0.3867110s)
- √ crowdsaleConstructor_EndTimePassed_ExpectRevert (0.0492880s)
- √ indexed - Event Transfer defines parameter 1 at index to of type address and indexed = True (0.0050560s)

Contract: HoshoAudit.CrowdsaleTests
- √ finalize_SaleOver_CrowdsaleFinalizedEvent (0.2520260s)
- √ buyTokens_BeneficiaryMadeReferrals_ProcessPurchase (0.6873350s)
- √ buyTokens_AllTokensLeftBonusPayout_AssertBonusPayout (0.6570910s)
- √ buyTokens_PurchaseMoreTokensThanInPhaseOne_AssertPayouts (0.5870760s)
- √ buyTokens_PurchaseMoreThanAvailable_ExpectRevert (0.2451010s)
- √ decreaseApproval_DecreaseByHalf_EmitEvent (0.1908360s)
- √ increaseApproval_Success (0.0418260s)
- √ finalize_SaleNotClosed_ExpectRevert (0.0379010s)
- √ finalize_BadTransfer_ExpectRevert (0.1708450s)
- √ fallbackFunction_buyingTokens_FromMsgSender (0.1212130s)
- √ crowdsaleConstructor_WalletIsAddressZero_ExpectRevert (0.1522460s)
- √ token_TokenERC20_ReturnToken (0.3783670s)
- √ owner_SenderIsOwner_AssertOwner (0.3683190s)
- √ indexed - Event Transfer defines parameter 2 at index value of type uint256 and indexed = False (0.0001770s)
- √ crowdsaleConstructor_StartTimeNotPassed_ExpectRevert (0.0312590s)
- √ decreaseApproval_DecreaseMoreThanAllowed_ExpectAllowanceSetTo0 (0.2051740s)
- √ buyTokens_FullBonusPayout_AssertBonusPayout (0.5605200s)
- √ crowdsaleConstructor_RateIsZero_ExpectRevert (0.1501790s)

Appendix A

Contract: HoshoAudit.CrowdsaleTests
√ allowance_CheckAmountApproved_AssertAreEqual (0.2864190s)
√ removeWhitelist_RemoveMultiple_AssertNotWhitelist (0.2232360s)
√ addAddressToWhitelist_SenderIsNotAdmin_ExpectRevert (0.0310720s)
√ whitelist_AddMultiple_AssertIsWhitelist (0.1171680s)
√ buyTokens_BonusIsZero_NoBonusDelivered (0.8163490s)
√ erc20_Basic_Standards (0.5025200s)

Contract: HoshoAudit.SafeMathOneTests
√ RevertAdditionOverflow (0.0043690s)
√ AllowRegularMultiply (0.0156140s)
√ RevertMultiplyOverflow (0.0140810s)
√ AllowRegularAddition (0.0177150s)
√ AllowRegularSubtraction (0.0033380s)

Contract: HoshoAudit.SafeMathOneTests
√ RevertSubtractionOverflow (0.0131330s)
√ SkipOperationMult0 (0.0152630s)
√ RevertDivideBy0 (0.0148950s)
√ mod_dividendIsZero_shouldRevert (0.0189410s)
√ mod_dividendisNotZero_shouldReturnCorrectValue (0.0208340s)
√ AllowRegularDivision (0.0157290s)
√ AllowRegularDivision (0.0614150s)
√ AllowRegularMultiply (0.0208310s)
√ AllowRegularAddition (0.0499740s)
√ RevertMultiplyOverflow (0.1584470s)
√ RevertAdditionOverflow (0.0529080s)
√ RevertDivideBy0 (0.0774940s)
√ AllowRegularSubtraction (0.0384190s)
√ mod_dividendIsZero_shouldRevert (0.0731720s)
√ mod_dividendisNotZero_shouldReturnCorrectValue (0.0802590s)
√ RevertSubtractionOverflow (0.0890620s)
√ SkipOperationMult0 (0.0270930s)
√ renounceOwnership_calledByOwner_EmitOwnershipTransferredEvent (0.1618880s)

Appendix A

Contract: HoshoAudit.TokenOwnableTests
- √ transferOwnership_newOwnerIsZeroAddress_ExpectRevert (0.0473650s)
- √ transferOwnership_senderIsNonOwner_ExpectRevert (0.0402460s)
- √ transferOwnership_calledByOwner_EmitOwnershipTransferredEvent (0.0887990s)
- √ transferOwnership_calledByOwner_UpdateOwnerVariable (0.1301200s)
- √ reclaimContract_GiveOwnershipBackToContract_EmitOwnershipTransferredEvent (0.2362410s)
- √ reclaimToken_TransferToNewOwnerAccount_AssertBalance (0.1521870s)

Contract: HoshoAudit.TokenTests
- √ transfer_SendMoreThanBalance_ExpectRevert (0.0497330s)
- √ decreaseApproval_DecreaseByHalf_EmitEvent (0.1925030s)
- √ tokenFallback_WillRevert_ExpectRevert (0.0390050s)
- √ decreaseApproval_Success (0.1708180s)
- √ totalSupply_CheckTotalSupply_AssertTotal (0.0361490s)

Contract: HoshoAudit.TokenTests
- √ transferFrom_ApproveThenTransfer_EmitEvent (0.1155520s)
- √ reclaimToken_FailingTransfer_ExpectRevert (0.1734610s)
- √ reclaimEther_NoEther_Sure (0.1058830s)
- √ decreaseApproval_DecreaseMoreThanAllowed_ExpectAllowanceSetTo0 (0.1934500s)
- √ transferFrom_valueGreaterThanBalance_ExpectRevert (0.0154410s)
- √ increaseApproval_Success (0.0547350s)
- √ transferFrom_valueGreatThanAllowed_ExpectRevert (0.2099720s)
- √ constructor_MsgValueIsZero_ExpectRevert (0.1003740s)
- √ transferFrom_ToAccountZero_ExpectRevert (0.0920000s)
- √ transfer_ToAddressZero_ExpectRevert (0.0560910s)
- √ allowance_CheckAmountApproved_AssertAreEqual (0.2088180s)
- √ balanceOf_CheckOwnerBalance_AssertEqual (0.0546620s)

Appendix B

FILE	FINGERPRINT
ClarityCrowdsale.sol	111557393BD15AEE9FF1F8E9C09F389D81533504AE0E8692051F674D91C348F4
ClarityToken.sol	A3EB60A42E25CE697428BF22E13C2056CC327B947C66CEE580A469C991C08016

Appendix C

FILE	% BRANCHES	% FUNCTION	% LINES
ClarityCrowdsale.sol	92.65%	94.34%	97.93%
ClarityToken.sol	90.91%	90.62%	94.19%
ALL FILES	91.96%* (103/112)	92.94%* (79/85)	96.54%* (223/231)

* Totals are calculated using weighted percentages