Quantstamp Contract Security Certificate

January 16th 2020 — Quantstamp Verified

GoodDollar

This smart contract audit was prepared by Quantstamp, the protocol for securing smart contracts.

Executive Summary

Type	Decentralized Autonomous Organization (DAO)
Auditors	Leonardo Passos, Senior Research Engineer Nadir Akhtar, Research Engineer Yohei Oka, Forward Deployed Engineer
Timeline	2019-10-30 through 2020-01-09
EVM	Byzantium
Languages	Solidity
Methods	Architecture Review, Unit Testing, Functional Testing, Computer-Aided Verification, Manual Review
Specification	None
Source Code	

Repository	Commit
GoodContracts	7a63e0f75bf75

Total Issues	**12**	(10 Resolved)
High Risk Issues	**1**	(1 Resolved)
Medium Risk Issues	0	(0 Resolved)
Low Risk Issues	**10**	(8 Resolved)
Informational Risk Issues	**1**	(1 Resolved)
Undetermined Risk Issues	0	(0 Resolved)

High Risk	The issue puts a large number of users' sensitive information at risk, or is reasonably likely to lead to catastrophic impact for client's reputation or serious financial implications for client and users.
Medium Risk	The issue puts a subset of users' sensitive information at risk, would be detrimental for the client's reputation if exploited, or is reasonably likely to lead to moderate financial impact.
Low Risk	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.
Unresolved	Acknowledged the existence of the risk, and decided to accept it without engaging in special efforts to control it.
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).
Resolved	Adjusted program implementation, requirements or constraints to eliminate the risk.

Changelog

- 2019-10-30 - Initial report (commit 85824c2870f10)
- 2020-01-09 - Diff report (commit 7a63e0f75bf75)

Overall Assessment

In our initial audit, we had found serious vulnerability vectors requiring prompt attention from the GoodDollar team. Additionally, we had found parts of the code that had little to no documentation. After the fixes following our report, all high risk vulnerabilities have been fixed and documentation has improved; two low risk vulnerabilities, however, have not yet been fixed. Our audit was restricted to the code of the GoodDollar project, excluding all third-party code use in this project (DAOStatck, Openzeppelin, etc).

Summary of Findings

ID	Description	Severity	Status
QSP-1	Fake Avatar Can Bypass ` onlyOwnerOrAvatar()	High	Resolved
QSP-2	Non-Whitelisted Address Can Send GoodDollars	Low	Resolved
QSP-3	Missing Logic in `checkEntitlement()`	Low	Resolved
QSP-4	Self Invitation Allowed	Low	Resolved
QSP-5	DID Overriding	Low	Resolved
QSP-6	Inconsistent Whitelisting Logic	Low	Resolved
QSP-7	Inconsistent Registering Logic	Low	Resolved
QSP-8	`SignUpBonus` Scheme Can End Before Period Ends	Low	Resolved
QSP-9	No Access Modifier(s) restrict `transferAccount`	Low	Unresolved
QSP-10	`awardUser` May Not Always Give Users Their Eligible Rewards	Low	Resolved
QSP-11	`setDay` Logic is Broken	Low	Unresolved
QSP-12	Unlocked Pragma	Informational	Resolved

Quantstamp Audit Breakdown

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

Toolset

The notes below outline the setup and steps performed in the process of this audit.

Setup

Tool Setup:

- Maian
- Truffle
- Ganache
- SolidityCoverage
- Oyente
- Mythril
- Truffle-Flattener
- Securify
- Slither

Steps taken to run the tools:

1. Installed Truffle: `npm install -g truffle`
2. Installed Ganache: `npm install -g ganache-cli`
3. Installed the solidity-coverage tool (within the project's root directory): `npm install --save-dev solidity-coverage`
4. Ran the coverage tool from the project's root directory: `./node_modules/.bin/solidity-coverage`
5. Flattened the source code using `truffle-flattener` to accommodate the auditing tools.
6. Installed the Mythril tool from Pypi: `pip3 install mythril`
7. Ran the Mythril tool on each contract: `myth -x path/to/contract`
8. Ran the Securify tool: `java -Xmx6048m -jar securify-0.1.jar -fs contract.sol`
9. Installed the Oyente tool from Docker: `docker pull luongnguyen/oyente`
10. Migrated files into Oyente (root directory): `docker run -v $(pwd):/tmp - it luongnguyen/oyente`
11. Ran the Oyente tool on each contract: `cd /oyente/oyente && python oyente.py /tmp/path/to/contract`
12. Cloned the MAIAN tool: `git clone --depth 1 https://github.com/MAIAN-tool/MAIAN.git maian`
13. Ran the MAIAN tool on each contract: `cd maian/tool/ && python3 maian.py -s path/to/contract contract.sol`
14. Installed the Slither tool: `pip install slither-analyzer`
15. Run Slither from the project directory `slither .`

Assessment

Findings

QSP-1 Fake Avatar Can Bypass ` onlyOwnerOrAvatar()

Severity: *High Risk*

Status: Resolved

File(s) affected: `dao/AvatarGuard.sol`

Description: An attacker can pass in a fake `Avatar` contract as the parameter for the `onlyOwnerOrAvatar()` modifier and bypass the check that the caller has to be the contract owner or the DAO avatar.

Exploit Scenario: Attacker deploys a fake `Avatar` contract s.t. its owner function is set to return the owner of the `GoodDollar` contract. This allows, for instance, the attacker to bypass the `onlyOwnerOrAvatar()` modifier and set the `feeRecipient` to any arbitrary address. In fact, **ALL** functions using the `onlyOwnerOrAvatar()` are vulnerable.

Recommendation: Replace `onlyOwnerOrAvatar` with `onlyOwner`; alternatively rework the avatar logic. For instance, `AvatarGuard` could keep a list of whitelisted addresses; then, `onlyOwnerOrAvatar()` would allow only the owner or addresses that have been whitelisted.

QSP-2 Non-Whitelisted Address Can Send GoodDollars

Severity: *Low Risk*

Status: Resolved

File(s) affected: `token/GoodDollar.sol`

Description: `transferAndCall` allows a non-whitelisted account to send Good Dollars, deviating from the `transfer` implementation.

Recommendation: Add the `onlyWhitelisted` modifier to `transferAndCall`.

QSP-3 Missing Logic in `checkEntitlement()`

Severity: *Low Risk*

Status: Resolved

File(s) affected: `dao/schemes/FixedUBI.sol`

Description: L33-39 and L59-63 have similar logic, but L59-63 does not have the `lastClaimed[user] = periodStart.sub(1 days)` logic (L34). This discrepancy will result in different results while trying to calculate the same number.

Recommendation: Keep logic consistent.
Update: while the provided fix (see commit 7a63e0f75bf75) solves the initial issue we pointed out, another issue occurs as a side effect. The `checkEntitlement` internal documentation states that it computes the claim amount even in cases of non-whitelisted users. However, the signature of the `claim` function says something different: a claim can only occur for whitelisted users. Thus, we argue that `checkEntitlement` should only be called for whitelisted users; as such, it should check that `identify.dateAdded[account]` different from zero (e.g., by placing a `require` statement). After that, the computation logic of the claim amount shall be the same for both `checkEntitlement` and `claim` functions.

QSP-4 Self Invitation Allowed

Severity: *Low Risk*

Status: Resolved

File(s) affected: `dao/schemes/InviteUser.sol`

Description: On L52, user can invite himself and earn double the expected amount of rewards after being whitelisted and executing `claimReward()`.

Recommendation: Check that invited user does not equal `msg.sender`.

QSP-5 DID Overriding

Severity: *Low Risk*

Status: Resolved

File(s) affected: `identity/Identity.sol`

Description: On L58, code cannot guarantee uniqueness with

```
bytes32 pHash = keccak256(bytes(did));
```

Thus, it is possible to override values in `didHashToAddress`. Any two addresses passed in with the same `did` string will cause an override.

Recommendation: If there is the expectation that `dids` are unique per address, make sure that only one did binds to any given address. For instance, by taking the hash of the `did` string with the input address; update all calculations of `pHash` accordingly.

QSP-6 Inconsistent Whitelisting Logic

Severity: *Low Risk*

Status: Resolved

File(s) affected: `identity/Identity.sol`

Description: `whitelistedCount` only increases/decreases for non-contract accounts. Consider that initially one has no whitelisted addresses, but then whitelists one contract address, say X. Then, calling `isWhitelisted(X)` returns true, but `getWhitelistedCount` returns zero, which seems inconsistent.

Recommendation: If `isWhitelisted(X)` returns true, then `X` should be counted as being whitelisted. If that is the case, then current implementation is incorrect; if correct, we advise having improved documentation and function naming to better reflect intent.

QSP-7 Inconsistent Registering Logic

Severity: *Low Risk*

Status: Resolved

File(s) affected: `dao/schemes/SchemeGuard.sol`

Description: Following `isRegistered`, `onlyRegistered` does not perform the same checks, i.e., (i) `avatar` is not `Avatar(0)`; (ii) `controller.isSchemeRegistered(address(this), address(avatar))`;

Recommendation: Make sure `isRegistered` and `onlyRegistered` rely on the same checks.

QSP-8 `SignUpBonus` Scheme Can End Before Period Ends

Severity: *Low Risk*

Status: Resolved

File(s) affected: `dao/schemes/SignUpBonus.sol`

Description: `end()` does not rely on the `requirePeriodEnd` modifier; thus, it is possible to end the scheme before its end period.

Recommendation: Add `requirePeriodEnd` to `end()`.
Update: as per clarification in code (see commit 7a63e0f75bf75), the `end` function deactivates the contract and transfers any remaining GoodDollar back to the avatar. `end()` can be called at any time by identity admins. Thus, we consider this issue as fixed.

QSP-9 No Access Modifier(s) restrict `transferAccount`

Severity: *Low Risk*

Status: Unresolved

File(s) affected: `identity/Identity.sol`

Description: `transferAccount` does not have any access modifier. Basically, a whitelisted address can transfer his account to a non-whitelisted/non-blacklisted account. Whitelisting an address should be the privilege of `onlyRegistered` and `onlyIdentityAdmin` users.

Recommendation: Use the same modifiers of `addWhitelistedWithDID` in `transferAccount`.

QSP-10 `awardUser` May Not Always Give Users Their Eligible Rewards

Severity: *Low Risk*

Status: Resolved

File(s) affected: `dao/schemes/InviteUser.sol`

Description: On L84-85, `newReward` is alway equal to zero. That occurs whenever `rewarded[_user]` (before the update on L84) is less than `maxBonus`, but `rewarded[_user] + reward > maxBonus`. In such cases, users won't be able to claim the `maxBonus - rewarded[_user]` reward they are eligible for.

Recommendation: Fix math logic.

QSP-11 `setDay` Logic is Broken

Severity: *Low Risk*

Status: Unresolved

File(s) affected: `contracts/dao/schemes/FixedUBI.sol`

Description: `setDay` does not work if called in intervals that occur in less than 24 hours. Since `lastCalc` is updated on L49, there is not much elapsed time to consider, causing `dayDiff / 1 days` to always be zero.

Recommendation: Calculate `currentDay` based on how much time has elapsed since `_periodStart`.
Update: We noticed a similar (unfixed) case in `AdminWallet.sol`; however, the effect there is minor. It suffices to rewrite the comment on L157 (commit 7a63e0f75bf75) "can only be done by admin the amount of times specified in constructor per day" as "can only be done by admin the amount of times specified in constructor per every 24 hour interval", as well as adjusting the error message on L162 to inform that it is a 24h interval (not per day).

QSP-12 Unlocked Pragma

Severity: *Informational*

Status: Resolved

File(s) affected: `wallet/AdminWallet.sol`, `token/GoodDollar.sol`, `identity/IdentityGuard.sol`, `Migrations.sol`

Description: Every Solidity file specifies in the header a version number of the format `pragma solidity (^)0.4.*`. 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." For consistency and to prevent unexpected behavior in the future, it is recommended to remove the caret to lock the file onto a specific Solidity version.

Adherence to Best Practices

• As a general practice, avoid shadowing variables, i.e., making use of function parameters whose names are the same as storage variables or views functions in the same contract or in inherited ones. For instance, `ReputationReward.avatar (dao/schemes/ReputationReward.sol#13)` shadows `SchemeGuard.avatar (dao/schemes/SchemeGuard.sol#13)`, `GoodDollar.decimals (token/GoodDollar.sol#14)` shadows `DAOToken.decimals (@daostack/arc/contracts/controller/DAOToken.sol#18)` **unresolved**, `ERC677Token.constructor.name (local variable @ token/ERC677Token.sol#10)` shadows `DAOToken.name (state variable @ @daostack/arc/contracts/controller/DAOToken.sol#15)` **resolved**, `ERC677Token.constructor.symbol (local variable @ token/ERC677Token.sol#11)` shadows `DAOToken.symbol (state variable @ @daostack/arc/contracts/controller/DAOToken.sol#16)` **resolved**, `ERC677Token.constructor.cap (local variable @ token/ERC677Token.sol#12)` shadows `DAOToken.cap (state variable @ @daostack/arc/contracts/controller/DAOToken.sol#19)` **resolved**, `GoodDollar.constructor.name (local variable @ token/GoodDollar.sol#24)` shadows `DAOToken.name (state variable @ @daostack/arc/contracts/controller/DAOToken.sol#15)` **resolved**, `GoodDollar.constructor.symbol (local variable @ token/GoodDollar.sol#25)` shadows `DAOToken.symbol (state variable @ @daostack/arc/contracts/controller/DAOToken.sol#16)` **resolved**, `GoodDollar.constructor.cap (local variable @ token/GoodDollar.sol#26)` shadows `DAOToken.cap (state variable @ @daostack/arc/contracts/controller/DAOToken.sol#19)` **resolved**, `GoodDollar.constructor.formula (local variable @ token/GoodDollar.sol#27)` shadows `FormulaHolder.formula (state variable @ dao/schemes/FormulaHolder.sol#10)` **resolved**, `GoodDollar.constructor.identity (local variable @ token/GoodDollar.sol#28)` shadows `IdentityGuard.identity (state variable @ identity/IdentityGuard.sol#14)` **resolved**

• Many dependencies in `package.json` are not locked. Lock them **unresolved:** see `devDependencies`.

• Be consistent and use `uint256` instead of `uint` **resolved**.

• `identity/Identity.sol`: there is no need to have an accessor function for `whitelistedCount`. Just make the latter public **resolved**.

• Some parts of code are poorly documented or have no documentation at all. At the very least, document all public/external functions **unresolved:** some functions still miss documentation; e.g., `dao/DaoCreator.sol` (function create).

• There are some typos in comments (e.g., in `Identity.sol`: Eligble, in `AdminWallet.sol`: byu, in `ReserveRelayer.sol`: forign, etc). Consider using a spellchecker (e.g., aspell) **unresolved**.

• When using a language visibility modifier (e.g., "public"), make sure it comes before any developer defined modifier(s) **unresolved**.

• `dao/schemes/ActivePeriod.sol`: in the constructor, document the time unit of `_periodStart` and `_periodEnd` **unresolved**.

• `dao/schemes/ActivePeriod.sol`: comment on lines 31-32 does not seem to convey what is in code. It says: “starts scheme if within period, gets avatar to add the address to list of admins and then ends even if still within period”, but there is no period control whatsoever. Adjust either the code or the comments to better reflect intent **resolved**.

• Whenever applicable, always check that input parameters are not-zero or non-null. We found that many constructors (e.g., in `ReserveMinter.sol`, `SignedUpBonus.sol`, `OneTimePayments.sol`, etc) do not follow this practice **unresolved:** some functions still don't check input parameters.

• `dao/schemes/ReputationReward.sol`: require message is misleading. It says "reputation reward cannot be equal to or lower than zero". However, the reputation reward will never be lower than zero, as the value parameter is `uint` **resolved**.

• `token/ERC677.sol`: declare ERC677 as an interface, as it has no behavior **resolved**.

• `token/ERC677Receiver.sol`: declare ERC677Receiver as an interface, as it has no behavior **resolved**.

• `token/ERC677Token.sol`: provide a string message to the require statement on L33 **resolved**.

• Many functions return a bool value, but are not wrapped inside a require statement when called---e.g, all calls to `controller.genericCall()`. Double check all such calls and when applicable, wrap them inside a require statement **unresolved**.

• `dao/DaoCreator.sol`: L4, 5, 6, 9 imported but unused in the contract **resolved**.

• `dao/DaoCreator.sol`: L33-37 missing documentation for most of the parameters **resolved**.

• `dao/DaoCreator.sol`: L41 `Reputation.sol` used but not imported in contract **resolved**.

• It is always advisable **NOT** to have side effects after external calls as a means to prevent re-entrancy. This practice is not followed by many functions we have inspected (e.g., `DaoCreatorGoodDollar._forgeOrg`, `InviteUser.claimReward()`, etc). The risk is mitigated if DAOStack, controllers, and schemes are trustworthy (which we assume they are) **unresolved**.

- `dao/schemes/UBI.sol`: switch the ordering of L155 and 157 to prevent potential re-entrancy **resolved**.

- `dao/schemes/UBI.sol`: L199 `claimers` may include users who joined between `periodStart` and when `start()` is called. These users will be counted as part of `claimers`, but will not be eligible to claim. This negatively effects the `distributionFormula` payout amount **resolved**.

- `identity/Identity.sol`: L64 `ContractRemoved` event is not fired **resolved**.

- `identity/Identity.sol`: L79, 203 The comment says these functions should revert but they do not **resolved**.

- `token/GoodDollar.sol`: L13 lack of consistency; in other contracts, storage variables don't have the underscore prefix while the function arguments do. Reversed in this case **resolved**.

- `token/GoodDollar.sol`: L27 missing param documentation regarding `formula` **resolved**.

Test Results

Test Suite Results

```
Contract: Integration - Active Period
    ✓ should correctly propose and register scheme (349884 gas)
    ✓ should not end inactive scheme
    ✓ should not allow starting twice (106642 gas)
    ✓ should unregister scheme (315169 gas)

  Contract: Integration - adding minter
    ✓ should not create scheme with zero address (272361 gas)
    ✓ should not add minter (25325 gas)
    ✓ should correctly propose and register scheme (349948 gas)
    ✓ should add minter (45494 gas)

  Contract: adminWallet
    ✓ should transfer to admins (21000 gas)
    ✓ should fill wallet (42080 gas)
    ✓ should not top admin list when empty (22355 gas)
    ✓ should add admins (135410 gas)
    ✓ should add admins (168695 gas)
    ✓ should top admins (81225 gas)
    ✓ should remove single admin (16499 gas)
    ✓ should allow admin to whitelist and remove whitelist (144256 gas)
    ✓ should not allow non-admin to whitelist and remove whitelist (127115 gas)
    ✓ should allow admin to blacklist and remove blacklist (76733 gas)
    ✓ should not allow non-admin to blacklist and remove blacklist (124291 gas)
    ✓ should not allow to top wallet if user balance is too high (45744 gas)
    ✓ should allow to top wallet (122410 gas)
    ✓ should not allow to top wallet more than three times (170766 gas)
    ✓ should not allow whitelisting and awarding before setting signup (24292 gas)
    ✓ should whitelist user (84129 gas)
    ✓ should award users without whitelisting (100088 gas)

  Contract: Integration - Claiming Reputation
    ✓ should not allow creation of scheme with zero or less reputation
    ✓ should correctly propose Rep scheme (601249 gas)
    ✓ should correctly register Rep scheme (119483 gas)
    ✓ should reward whitelisted and creator for starting period (210182 gas)
    ✓ should reward for ending Rep period (172127 gas)

  Contract: Integration - rewarding whitelisted bonus
    ✓ should not allow signup with zero max bonus (589293 gas)
    ✓ should not allow awarding before starting scheme (23570 gas)
    ✓ should start SignUpBonus scheme (573331 gas)
    ✓ should not allow awarding by non admin (26366 gas)
    ✓ should allow awarding (93396 gas)
    ✓ should not allow awarding more than max bonus (29517 gas)
    ✓ should end SignUpBonus scheme (70409 gas)
    ✓ should unregister SignUpBonus scheme (315233 gas)
    ✓ should start empty SignUpBonus scheme (487576 gas)
    ✓ should not start empty SignUpBonus scheme (486943 gas)
    ✓ should end empty SignUpBonus scheme (56102 gas)

  Contract: Integration - Claiming UBI
    ✓ should not allow creating fixed UBI contract with zero distribution (675744 gas)
    ✓ should allow non-whitelisted to checkEntitlement
    ✓ should end UBI scheme with no remaining reserve (607645 gas)
    ✓ should perform transactions and increase fee reserve (94935 gas)
    ✓ should correctly propose UBI scheme (230465 gas)
    ✓ should correctly register UBI scheme (119483 gas)
    ✓ should start UBI period (239355 gas)
    ✓ should not allow starting scheme without enough funds (492353 gas)
    ✓ should register fixed claim scheme and add whitelisteds (501992 gas)
    ✓ should correctly claim UBI (144746 gas)
    ✓ should show amount of whitelisteds
    ✓ should show amount claimed
    ✓ should not allow to claim twice (28058 gas)
    ✓ should not allow non-whitelisted to claim (24665 gas)
    ✓ should not allow new whitelisted to claim (103514 gas)
    ✓ should end UBI period (100894 gas)
    ✓ should allow starting fixed claim scheme (143042 gas)
    ✓ should correctly register ReserveRelayer scheme and transfer new fees to fixed UBI (735560 gas)
    ✓ should allow claiming from fixed UBI (149160 gas)
    ✓ should not allow claiming on same day from fixed UBI (25787 gas)
    ✓ should not allow to claim for more than seven days (164026 gas)
    ✓ should get daily stats

  Contract: Dao - Forging organizations, adding founders
    ✓ should not allow to forge organizations twice (7538028 gas)
    ✓ should not allow stranger to set schemes (28102 gas)
    ✓ should not allow zero address founder (4046860 gas)
    ✓ should not allow founders without reputation to forge org (32470 gas)
```

```
    ✓ should not allow founders without tokens to forge org (32441 gas)
    ✓ should not allow sender to forge org without founders (30436 gas)
    ✓ should not mint reputation or tokens to zero address founder (4188013 gas)

Contract: ERC677 token
    ✓ Should not allow setting non contract as bridge contract (70181 gas)
    ✓ Should allow setting contract as bridge contract (44771 gas)

Contract: FeeFormula - setting transaction fees
    ✓ should not allow FormulaHolder with null formula (143422 gas)
    ✓ should not allow Fee formula with too high percentage (198295 gas)
    ✓ should be allowed to register new formula (416057 gas)
    ✓ should not allow stranger to change formula (23335 gas)
    ✓ should allow owner to set new formula (34284 gas)
    ✓ should have support 0 tx fee
    ✓ should calculate tx fee correctly

Contract: GoodDollar
    ✓ should fail transfer (39518 gas)
    ✓ should transfer and not call function (128696 gas)
    ✓ should transfer, not call and return true if not contract (69832 gas)
    ✓ should transfer and call correct function on receiver contract (149407 gas)
    ✓ should increase allowance (52358 gas)
    ✓ should allow to transfer from (85278 gas)
    ✓ should decrease allowance (22281 gas)
    ✓ should allow to burn (37678 gas)
    ✓ should allow to burn from (86927 gas)
    ✓ should not allow to mint beyond cap (97848 gas)

Contract: Identity - Blacklist and whitelist
    ✓ should set avatar (66293 gas)
    ✓ should blacklist address (69735 gas)
    ✓ should check blacklisted (229331 gas)
    ✓ should add, check and remove whitelisted (160278 gas)
    ✓ should increment and decrement whitelisteds when adding whitelisted (108530 gas)
    ✓ should revert when non admin tries to add whitelisted (27591 gas)
    ✓ should revert when non admin tries to add blacklist (27613 gas)
    ✓ should not be able to add zero address as identity admin
    ✓ should add identity admin (1550645 gas)
    ✓ should remove identity admin (1107452 gas)
    ✓ should not remove identity admin twice (282334 gas)
    ✓ should renounce identity admin (1246862 gas)
    ✓ should revert when adding to whitelisted twice (136808 gas)
    ✓ should revert when adding to blacklist twice (98017 gas)
    ✓ should not increment whitelisted counter when adding whitelisted (136808 gas)
    ✓ should renounce whitelisted (104342 gas)
    ✓ should add with did (119049 gas)
    ✓ should not allow adding with used did (29678 gas)
    ✓ should not allow transferring account to blacklisted (26426 gas)
    ✓ should not allow transferring account to address with funds (123905 gas)
    ✓ should not allow transferring account to address with did (148990 gas)
    ✓ should transfer account to new address (135345 gas)
    ✓ should not keep did after transferring account
    ✓ should not allow setting non-registered identity contract (2274670 gas)
    ✓ should allow to set registered identity (34332 gas)
    ✓ should not allow adding non contract to contracts (28541 gas)

Contract: Integration - awarding invitational bonus
    ✓ should not allow creating scheme with higher reward than max (415324 gas)
    ✓ should not allow claiming before starting scheme (25487 gas)
    ✓ should start InviteUser scheme (349948 gas)
    ✓ Should invite user (50481 gas)
    ✓ Should not be able to invite self (30213 gas)
    ✓ should not allow inviting twice (27311 gas)
    ✓ should not allow inviting registered whitelisted (30113 gas)
    ✓ should allow registered whitelisted to claim (112396 gas)
    ✓ should not allow whitelisted to claim twice (28575 gas)
    ✓ should allow to claim after registering (250102 gas)
    ✓ should end InviteUser scheme (315233 gas)

Contract: ReserveMinter - Minting to reserve
    ✓ should not allow relayer with null address receiver
    ✓ should not allow relayer with zero to transfer (753361 gas)
    ✓ should correctly propose ReserveMinter scheme (230465 gas)
    ✓ should correctly register ReserveMinter scheme (119483 gas)
    ✓ should start, mint to receiver and then end (69591 gas)

  Contract: Integration - One-Time Payments
    ✓ should not allow One-Time payments before registering (187572 gas)
    ✓ should correctly propose One-Time Payment scheme (230465 gas)
    ✓ should correctly register One-Time payment scheme (243392 gas)
    ✓ should not have payment
    ✓ should only allow token to deposit
    ✓ should deposit successfully (222999 gas)
    ✓ should not allow to deposit to same hash (77777 gas)
    ✓ should have payment
    ✓ should not withdraw with wrong signature (37046 gas)
    ✓ should withdraw successfully (62843 gas)
    ✓ should not allow withdraw from unused link (38020 gas)
    ✓ should not allow to withdraw from already withdrawn (38084 gas)
    ✓ should only allow creator of deposit to cancel (250192 gas)
    ✓ should propose to unregister One-Time payment scheme (203579 gas)
    ✓ should correctly unregister One-Time payment scheme (111654 gas)
    ✓ should not allow One-Time payments after registering (157340 gas)
    ✓ should remove oneTimePayments from whitelisted without decrementing amount of whitelisted non contracts
(37726 gas)

  Contract: Ownership - transferring ownership to controller
    ✓ fee formula should have proper owner
    ✓ identity should have proper owner

  Contract: ReserveRelayer - Transferring reserve
    ✓ should not allow relayer with null address receiver
```

```
    ✓ should perform transactions and increase fee reserve (224805 gas)
    ✓ should correctly propose ReserveRelayer scheme (230465 gas)
    ✓ should correctly register ReserveRelayer scheme (119483 gas)
    ✓ should start, transfer reserve and then end (213788 gas)

  Contract: SchemeGuard - registered schemes
    ✓ should not start scheme (55879 gas)
    ✓ should correctly propose and register scheme (349948 gas)
    ✓ should have registered scheme
    ✓ should start scheme (25313 gas)
    ✓ should not end scheme (25358 gas)
    ✓ should unregister scheme (315233 gas)
    ✓ should end scheme (25342 gas)

·-----------------------------------------------|---------------------------|-------------|-----------------
-----------------·
|      Solc version: 0.5.4+commit.9549d8ff       ·  Optimizer enabled: true  ·  Runs: 200  ·  Block limit:
17592186044415 gas  |
·-----------------------------------------------|---------------------------|-------------|
·---------------------------------·
|  Methods                                       ·               1 gwei/gas                ·          138.00
usd/eth           |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------·
|  Contract              ·  Method                ·  Min        ·  Max        ·  Avg        ·  # calls
·  usd (avg)     |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  ActivePeriod          ·  end                   ·      17370  ·      77554  ·      56930  ·              10
·          0.01  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  ActivePeriod          ·  start                 ·      43839  ·     190967  ·     153543  ·               5
·          0.02  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  AddAdmin              ·  start                 ·      43224  ·     188112  ·      77475  ·               5
·          0.01  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  AddAdmin              ·  transferOwnership     ·          -  ·          -  ·      30594  ·               2
·          0.00  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  AddMinter             ·  addMinter             ·          -  ·          -  ·      45494  ·               2
·          0.01  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  AdminWallet           ·  addAdmins             ·     135410  ·     168695  ·     152053  ·               4
·          0.02  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  AdminWallet           ·  blacklist             ·          -  ·          -  ·      53424  ·               2
·          0.01  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  AdminWallet           ·  removeAdmins          ·          -  ·          -  ·      16499  ·               2
·          0.00  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  AdminWallet           ·  removeBlacklist       ·          -  ·          -  ·      23309  ·               4
·          0.00  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  AdminWallet           ·  removeWhitelist       ·          -  ·          -  ·      35975  ·               2
·          0.00  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  AdminWallet           ·  topAdmins             ·          -  ·          -  ·      81225  ·               2
·          0.01  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  AdminWallet           ·  topWallet             ·      41366  ·     101370  ·      61367  ·               3
·          0.01  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  AdminWallet           ·  whitelist             ·      79555  ·     108281  ·      93918  ·               2
·          0.01  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  AdminWallet           ·  whitelistAndAwardUser ·      84129  ·     100088  ·      89449  ·               3
·          0.01  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  DaoCreatorGoodDollar  ·  forgeOrg              ·          -  ·          -  ·    7505866  ·               1
·          1.04  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  FeeFormula            ·  setAvatar             ·          -  ·          -  ·      66109  ·               2
·          0.01  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  FixedUBI              ·  claim                 ·     141412  ·     149160  ·     145286  ·               4
·          0.02  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  FixedUBI              ·  start                 ·          -  ·          -  ·     143042  ·               2
·          0.02  |
·----------------------|-----------------------|-------------|-------------|-------------|
·----------------|-----------------·
|  FormulaHolderMock     ·  setFormula            ·          -  ·          -  ·      34284  ·               3
·          0.00  |
·----------------------|-----------------------|-------------|-------------|-------------|
```

GoodDollar	approve	-	-	51795	1	0.01
GoodDollar	burn	22614	37678	32657	3	0.00
GoodDollar	burnFrom	-	-	35132	2	0.00
GoodDollar	decreaseAllowance	-	-	22281	2	0.00
GoodDollar	increaseAllowance	-	-	52358	2	0.01
GoodDollar	mint	-	-	70014	1	0.01
GoodDollar	setBridgeContract	-	-	44771	1	0.01
GoodDollar	transfer	64871	94935	75922	15	0.01
GoodDollar	transferAndCall	69832	158064	126096	7	0.02
GoodDollar	transferFrom	-	-	85278	2	0.01
Identity	addBlacklisted	-	-	49892	4	0.01
Identity	addWhitelisted	-	-	76022	20	0.01
Identity	addWhitelistedWithDID	119049	119113	119070	3	0.02
Identity	removeBlacklisted	-	-	19843	6	0.00
Identity	removeWhitelisted	32508	37726	33088	9	0.00
Identity	renounceIdentityAdmin	-	-	14377	2	0.00
Identity	renounceWhitelisted	-	-	28320	2	0.00
Identity	setAvatar	-	-	66293	2	0.01
Identity	transferAccount	-	-	135345	3	0.02
IdentityGuardMock	blacklistMock	-	-	28793	1	0.00
IdentityGuardMock	checkWhitelisted	-	-	25827	1	0.00
IdentityGuardMock	setIdentity	-	-	34332	2	0.00
InviteUser	claimReward	112396	145799	129098	4	0.02
InviteUser	inviteUser	-	-	50481	2	0.01
Migrations	setCompleted	-	-	26929	2	0.00
OneTimePayments	cancel	-	-	33595	2	0.00

OneTimePayments	start	-	-	123909	4
0.02					
OneTimePayments	withdraw	-	-	62843	2
0.01					
RemoveAdmin	start	-	-	26108	2
0.00					
RemoveAdmin	transferOwnership	-	-	30594	1
0.00					
ReputationMock	start	-	-	188307	1
0.03					
SchemeGuardMock	end	-	-	25342	1
0.00					
SchemeGuardMock	start	-	-	25313	2
0.00					
SchemeGuardMock	transferOwnership	-	-	30530	1
0.00					
SignUpBonus	awardUser	-	-	93396	2
0.01					
SignUpBonus	start	137724	143512	140618	4
0.02					
UBI	claim	-	-	144746	4
0.02					
UBI	start	-	-	213664	2
0.03					

Deployments			% of limit
ActivePeriodMock	-	-	326095	0 %
0.05				
AddAdmin	-	-	808751	0 %
0.11				
AddFoundersGoodDollar	-	-	2011066	0 %
0.28				
AddMinter	-	-	692910	0 %
0.10				
AdminWallet	1485512	1550678	1518095	0 %
0.21				
ControllerCreatorGoodDollar	-	-	4467243	0 %
0.62				
DaoCreatorGoodDollar	-	-	5105715	0 %
0.70				
FeeFormula	-	-	498067	0 %
0.07				
FixedUBI	-	-	2155977	0 %
0.30				
FormulaHolderMock	-	-	332552	0 %
0.05				
Identity	-	-	2245482	0 %
0.31				
IdentityGuardMock	-	-	567096	0 %
0.08				
Migrations	-	-	221171	0 %

```
·          0.03  |
·····················|·····|·····|·····|·····|·····|·····|·····|·····|
|  OneTimePayments                                  ·     -  ·     -  ·  1439227  ·          0 %
·          0.20  |
·····················|·····|·····|·····|·····|·····|·····|·····|·····|
|  RemoveAdmin                                      ·     -  ·     -  ·   700802  ·          0 %
·          0.10  |
·····················|·····|·····|·····|·····|·····|·····|·····|·····|
|  ReserveRelayer                                   ·     -  ·     -  ·  1343289  ·          0 %
·          0.19  |
·····················|·····|·····|·····|·····|·····|·····|·····|·····|
|  SchemeGuardMock                                  ·     -  ·     -  ·   523567  ·          0 %
·          0.07  |
·····················|·····|·····|·····|·····|·····|·····|·····|·····|
|  SignUpBonus                                      ·     -  ·     -  ·  1835528  ·          0 %
·          0.25  |
·-----------------------------------------------|-----------|-----------|-----------|-----------------
|-----------------·
```

Code Coverage

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Lines
dao/	100	100	100	100	
DaoCreator.sol	100	100	100	100	
dao/schemes/	100	100	100	100	
ActivePeriod.sol	100	100	100	100	
AddAdmin.sol	100	100	100	100	
AddMinter.sol	100	100	100	100	
FeeFormula.sol	100	100	100	100	
FeelessScheme.sol	100	100	100	100	
FixedUBI.sol	100	100	100	100	
FormulaHolder.sol	100	100	100	100	
InviteUser.sol	100	100	100	100	
OneTimePayments.sol	100	100	100	100	
RemoveAdmin.sol	100	100	100	100	
ReputationReward.sol	100	100	100	100	
ReserveMinter.sol	100	100	100	100	
ReserveRelayer.sol	100	100	100	100	
SchemeGuard.sol	100	100	100	100	
SignUpBonus.sol	100	100	100	100	
UBI.sol	100	100	100	100	
identity/	100	100	100	100	
Identity.sol	100	100	100	100	
IdentityAdminRole.sol	100	100	100	100	
IdentityGuard.sol	100	100	100	100	
mocks/	100	100	100	100	
ActivePeriodMock.sol	100	100	100	100	
BridgeMock.sol	100	100	100	100	
FormulaHolderMock.sol	100	100	100	100	
IdentityGuardMock.sol	100	100	100	100	
ReputationMock.sol	100	100	100	100	
SchemeGuardMock.sol	100	100	100	100	
TransferAndCallMock.sol	100	100	100	100	
token/	100	100	100	100	
ERC677BridgeToken.sol	100	100	100	100	
ERC677Token.sol	100	100	100	100	
GoodDollar.sol	100	100	100	100	
token/ERC677/	100	100	100	100	
ERC677.sol	100	100	100	100	
ERC677Receiver.sol	100	100	100	100	
wallet/	100	100	100	100	
AdminWallet.sol	100	100	100	100	
----------	---------	---------	---------	---------	----------
All files	**100**	**100**	**100**	**100**	
----------	---------	---------	---------	---------	----------

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Lines
dao/	100	100	100	100	
AvatarGuard.sol	100	100	100	100	
DaoCreator.sol	100	100	100	100	
dao/schemes/	100	100	100	100	
ActivePeriod.sol	100	100	100	100	
AddAdmin.sol	100	100	100	100	
AddMinter.sol	100	100	100	100	
FeeFormula.sol	100	100	100	100	
FeelessScheme.sol	100	100	100	100	
FixedUBI.sol	100	100	100	100	
FormulaHolder.sol	100	100	100	100	
InviteUser.sol	100	100	100	100	
OneTimePayments.sol	100	100	100	100	
RemoveAdmin.sol	100	100	100	100	
ReputationReward.sol	100	100	100	100	
ReserveMinter.sol	100	100	100	100	
ReserveRelayer.sol	100	100	100	100	
SchemeGuard.sol	100	100	100	100	
SignUpBonus.sol	100	100	100	100	
UBI.sol	100	100	100	100	
identity/	100	100	100	100	
Identity.sol	100	100	100	100	
IdentityAdminRole.sol	100	100	100	100	
IdentityGuard.sol	100	100	100	100	
mocks/	100	100	88.24	100	
ActivePeriodMock.sol	100	100	100	100	
FormulaHolderMock.sol	100	100	0	100	
IdentityGuardFailMock.sol	100	100	0	100	
IdentityGuardMock.sol	100	100	100	100	
ReputationMock.sol	100	100	100	100	
SchemeGuardMock.sol	100	100	100	100	
TransferAndCallMock.sol	100	100	100	100	
token/	100	100	100	100	
ERC677Token.sol	100	100	100	100	
GoodDollar.sol	100	100	100	100	
token/ERC677/	100	100	100	100	
ERC677.sol	100	100	100	100	
ERC677Receiver.sol	100	100	100	100	
wallet/	84	56.25	93.33	86.11	
AdminWallet.sol	84	56.25	93.33	86.11	… 164,167,168
All files	**98.71**	**95.27**	**98.09**	**98.79**	

About Quantstamp

Quantstamp is a Y Combinator-backed company that helps to secure smart contracts at scale using computer-aided reasoning tools, with a mission to help boost adoption of this exponentially growing technology.

Quantstamp's team boasts decades of combined experience in formal verification, static analysis, and software verification. Collectively, our individuals have over 500 Google scholar citations and numerous published papers. In its mission to proliferate development and adoption of blockchain applications, Quantstamp is also developing a new protocol for smart contract verification to help smart contract developers and projects worldwide to perform cost-effective smart contract security audits.

To date, Quantstamp has helped to secure hundreds of millions of dollars of transaction value in smart contracts and has assisted dozens of blockchain projects globally with its white glove security auditing services. As an evangelist of the blockchain ecosystem, Quantstamp assists core infrastructure projects and leading community initiatives such as the Ethereum Community Fund to expedite the adoption of blockchain technology.

Finally, Quantstamp's dedication to research and development in the form of collaborations with leading academic institutions such as National University of Singapore and MIT (Massachusetts Institute of Technology) reflects Quantstamp's commitment to enable world-class smart contract innovation.