SOLIDIFIED

Decentralized Audit Platform and Bug Prediction Market

Whitepaper

IMPORTANT LEGAL NOTICE

PLEASE READ THIS SECTION AND ALL THE FOLLOWING SECTIONS CAREFULLY. IF YOU ARE IN ANY DOUBT AS TO THE ACTION YOU SHOULD TAKE, YOU SHOULD CONSULT YOUR LEGAL, FINANCIAL, TAX OR OTHER PROFESSIONAL ADVISOR(S).

This whitepaper does not constitute an invitation or inducement to engage in any investment activity nor form part of any advice to sell, or any solicitation of any offer, by Solidified Ltd ("Solidified") or any third party (as applicable, the "Distributor") to purchase any tokens of Solidified, nor shall it or any part of it, nor the fact of its presentation, form the basis of, or be relied upon in connection with, any contract or investment decision.

No person is bound to enter into any contract or binding legal commitment in relation to the sale and purchase of the tokens, and no cryptocurrency or other form of payment is to be accepted, on the basis of this whitepaper alone. Any agreement between the Distributor and you in relation to any sale and purchase of Solid Tokens (as referred to in this whitepaper) is to be governed only by separate documents, such as a token sale agreement and other applicable terms and conditions, setting out the terms and conditions of such agreement (the "Agreement") .

This whitepaper is for information purposes only and is subject to change. Solidified cannot guarantee the accuracy of the statements made or conclusions reached in this whitepaper. Solidified does not make and expressly disclaims all representations and warranties (whether express or implied by statute or otherwise) whatsoever, including but not limited to:

- any representations or warranties relating to merchantability, fitness for a particular purpose, suitability, wage, title or non-infringement;
- that the contents of this whitepaper are accurate and free from any errors; and
- that such contents do not infringe any third party rights. Solidified shall have no liability for damages of any kind arising out of the use, reference to or reliance on the contents of this whitepaper, even if advised of the possibility of such damages.

In the event of any inconsistencies between the Agreement and this whitepaper, the Agreement shall prevail.

Without prejudice to any other limitations set out in the Agreement, you are not eligible and you are not to purchase any Solid Tokens if you are a citizen or resident (tax or otherwise) of any jurisdiction in which the offer and sale of the Solid Tokens is prohibited.

No regulatory authority has examined or approved of any of the information set out in this whitepaper. No such action has been or will be taken under the laws, regulatory requirements or rules of any jurisdiction. The publication, distribution or dissemination of this whitepaper does not imply that the applicable laws, regulatory requirements or rules have been complied with. Any Solid Tokens could be impacted by regulatory action, including potential restrictions on the ownership, use, or possession of such tokens. Regulators or other circumstances may demand that the mechanics of the Solid Tokens be altered, all or in part. Solidified may revise mechanics to comply with regulatory requirements or other governmental or business obligations.

This whitepaper, any part thereof and any copy thereof must not be taken or transmitted to any country where distribution or dissemination of this whitepaper is prohibited or restricted. No part of this whitepaper is to be reproduced, distributed or disseminated without including this section.

Please note that Solidified is in the process of undertaking a legal and regulatory analysis of the functionality of its Solid Tokens. Following the conclusion of this analysis, Solidified may decide to amend the intended functionality of its Solid Tokens in order to ensure compliance with any legal or regulatory requirements to which it is subject. In the event that Solidified decides to amend the intended functionality of its Solid Tokens, Solidified will update the relevant contents of this whitepaper and upload the latest version of this to its website.

SOLID TOKENS

As of the date of publication of this paper, the Solid Tokens have no known potential uses outside of the Solidified ecosystem, and are not permitted to be sold or otherwise traded on third-party exchanges. All proceeds received by Solidified from the sale of Solidified tokens may be spent freely by Solidified absent any conditions, save as set out herein.

CAUTIONARY NOTE ON FORWARD-LOOKING STATEMENTS

All statements contained in this whitepaper, statements made in press releases or in any place accessible by the public and oral statements that may be made by Solidified or its directors, executive officers, employees or agents acting on its behalf, that are not statements of historical fact, constitute "forward-looking statements". Some of these statements can be identified by forward-looking terms such as "aim", "target", "anticipate", "believe", "could", "estimate", "expect", "if", "intend", "may", "plan", "possible", "probable", "project", "should", "would", "will" or other similar terms. However, these terms are not the exclusive means of identifying forward-looking statements. All statements regarding Solidified's financial position, business strategies, plans and prospects and the future prospects of the industry which Solidified is in are forward-looking statements. These forward-looking statements, including but not limited to statements as to Solidified's revenue and profitability, prospects, future plans, other expected industry trends and other matters discussed in this whitepaper regarding Solidified are matters that are not historic facts, but only predictions.

These forward-looking statements involve known and unknown risks, uncertainties and other factors that may cause the actual future results, performance or achievements of Solidified to be materially different from any future results, performance or achievements expected, expressed or implied by such forward-looking statements. These factors include, amongst others:

(a) *changes in political, social, economic and stock or cryptocurrency market conditions, and the regulatory environment in the countries in which Solidified conducts its respective businesses and operations;*

(b) *the risk that Solidified may be unable to execute or implement its business strategy and future plans;*

(c) *changes in interest rates and exchange rates of fiat currencies and cryptocurrencies;*

(d) *changes in the anticipated growth strategies and expected internal growth of Solidified;*

(e) *changes in the availability and fees payable to Solidified in connection with its business and operations;*

(f) *changes in the availability and salaries of employees who are required for Solidified to operate its business and operations;*

(g) *changes in preferences of customers of Solidified;*

(h) *changes in competitive conditions under which Solidified operate, and the ability of Solidified to compete under such conditions;*

(i) *changes in the future capital needs of Solidified and the availability of financing and capital to fund such needs;*

(j) *war or acts of international or domestic terrorism;*

(k) *occurrences of catastrophic events, natural disasters and "acts of God" that affect the businesses and/or operations of Solidified;*

(l) *other factors beyond the control of Solidified; and*

(m) *any other risks and uncertainties associated with Solidified and its business and operations, the Solid Tokens and the sale of Solid Tokens.*

All forward-looking statements made by or attributable to Solidified or persons acting on behalf of Solidified are expressly qualified in their entirety by such factors. Given that risks and uncertainties that may cause the actual future results, performance or achievements of Solidified to be materially different from that expected, expressed or implied by the forward-looking statements in this whitepaper, undue reliance must not be placed on these statements. These forward-looking statements are applicable only as of the date of this whitepaper.

Neither Solidified nor any other person represents, warrants and/or undertakes that the actual future results, performance or achievements of Solidified will be as discussed in those forward looking statements. The actual results, performance or achievements of Solidified may differ materially from those anticipated in these forward-looking statements. Nothing contained in this whitepaper is or may be relied upon as a promise, representation or undertaking as to the future performance or policies of Solidified. Further, Solidified disclaims any responsibility to update any of those forward-looking statements or publicly announce any revisions to those forward-looking statements to reflect future developments, events or circumstances, even if new information becomes available or other events occur in the future.

RISKS AND UNCERTAINTIES

Prospective purchasers of Solid Tokens (as referred to in this whitepaper) should carefully consider and evaluate all risks and uncertainties associated with Solidified and its businesses and operations, the Solid Tokens and the sale of Solid Tokens, all risks, uncertainties and other information set out in the Agreement and this whitepaper, prior to any purchase of Solid Tokens. If any of such risks and uncertainties develops into actual events, the business, financial condition, results of operations and prospects of Solidified could be materially and adversely affected. In such cases, you may lose all or part of the value of the Solid Tokens. Solidified's business is subject to various laws and regulations in the countries where it operates or intends to operate. There is a risk that certain activities of Solidified may be deemed in violation of any such law or regulation. Penalties for any such potential violation would be unknown. Additionally, changes in applicable laws or regulations or evolving interpretations of existing law could, in certain circumstances, result in increased compliance costs or capital expenditures, which could affect Solidified's profitability, or impede Solidified's ability to carry on the business model and the Solid Tokens model proposed in this whitepaper. For any assistance on the assessment of such risks, you are invited to consult your legal, financial, tax or other professional advisors.

Solid Tokens should not be acquired for speculative or investment purposes with the expectation of making a profit or immediate re-sale. No promises of future performance or value are or will be made with respect to Solid Tokens, including no promise of inherent value, no promise of continuing payments, and no guarantee that Solid Tokens will hold any particular value. Unless prospective participants fully understand and accept the nature of Solidified and the potential risks inherent in Solid Tokens, they should not participate in the sale.

MARKET AND INDUSTRY INFORMATION AND NO CONSENT OF OTHER PERSONS

This whitepaper includes market and industry information and forecasts that have been obtained from internal surveys, reports and studies, where appropriate, as well as market research, publicly available information and industry publications. Such surveys, reports, studies, market research, publicly available information and publications generally state that the information that they contain has been obtained from

sources believed to be reliable, but there can be no assurance as to the accuracy or completeness of such included information.

Save for Solidified and its directors, executive officers and employees, no person has provided his or her consent to the inclusion of his or her name and/or other information attributed or perceived to be attributed to such person in connection therewith in this whitepaper and no representation, warranty or undertaking is or purported to be provided as to the accuracy or completeness of such information by such person and such persons shall not be obliged to provide any updates on the same.

While Solidified has taken reasonable actions to ensure that the information is extracted accurately and in its proper context, it has not conducted any independent review of the information extracted from third party sources, verified the accuracy or completeness of such information or ascertained the underlying economic assumptions relied upon therein. Consequently, neither Solidified , nor its directors, executive officers and employees acting on its behalf makes any representation or warranty as to the accuracy or completeness of such information and shall not be obliged to provide any updates on the same.

TERMS USED

To facilitate a better understanding of the Solid Tokens being offered for purchase by the Distributor, and the businesses and operations of Solidified, certain technical terms and abbreviations, as well as, in certain instances, their descriptions, have been used in this whitepaper. These descriptions and assigned meanings should not be treated as being definitive of their meanings and may not correspond to standard industry meanings or usage.

Words importing the singular shall, where applicable, include the plural and vice versa and words importing the masculine gender shall, where applicable, include the feminine and neuter genders and vice versa. References to persons shall include corporations and other business organizations.

NO FURTHER INFORMATION OR UPDATE

No person has been or is authorized to give any information or representation not contained in this whitepaper in connection with Solidified and its business and operations, the Solid Tokens and the sale of Solid Tokens and, if given, such information or representation must not be

relied upon as having been authorized by or on behalf of Solidified. The sale of Solid Tokens shall not, under any circumstances, constitute a continuing representation or create any suggestion or implication that there has been no change, or development reasonably likely to involve a material change in the affairs, conditions and prospects of Solidified or in any statement of fact or information contained in this whitepaper since the date hereof. Accordingly, this whitepaper is subject to change.

DISCLAIMER OF LIABILITY

To the maximum extent permitted by the applicable laws, regulations and rules, Solidified shall not be liable for any indirect, special, incidental, consequential or other losses of any kind, in tort, contract or otherwise (including but not limited to loss of revenue, income or profits, and loss of use or data), arising out of or in connection with any reliance on this whitepaper, or acceptance of the T&Cs, or any part thereof by you.

RESTRICTIONS ON DISTRIBUTION AND DISSEMINATION

The distribution or dissemination of this whitepaper or any part thereof may be prohibited or restricted by the laws, regulatory requirements and rules of any jurisdiction. In the case where any restriction applies, you are to inform yourself about, and to observe, any restrictions which are applicable to your possession of this whitepaper or such part thereof (as the case may be) at your own expense and without liability to Solidified. Persons to whom a copy of this whitepaper has been distributed or disseminated, provided access to or who otherwise have the whitepaper in their possession shall not circulate it to any other persons, reproduce or otherwise distribute this whitepaper or any information contained herein for any purpose whatsoever nor permit or cause the same to occur.

1 Introduction

Within the past 2 years, smart contract exploits have cost Ethereum users over €200,000,000 in Ether and other cryptoassets. This has lead the Ethereum community to highly value security audits, as a vital step in developing and deploying a smart contract[1]. We, as auditors and bug hunters, encourage this trend and strongly believe this stance is a net benefit for the Ethereum ecosystem; however, as auditors and bug hunters, we recognize that the smart contract audit market is flawed. The supply of qualified auditors remains greatly outpaced by the demand for smart contract audits; because of this, the cost of an audit is extremely high and the availability of top auditors is very low. Clients are expected to pay tens of thousands of euros (at minimum), on top of waiting many weeks for their contracts to even begin being audited.

[1] https://medium.com/new-alchemy/a-short-history-of-smart-contract-hacks-on-ethereum-1a30020b5fd

The most prominent audit firms are often booked solid for months, and for good reason, auditors have very limited throughput and attempts to increase it likely come at a cost of diminished diligence. Clients often have limited ability to judge the quality of an auditor's work, and therefore must (i) hire audit firms with only a rough understanding of their reputation, (ii) trust the auditor to perform their work competently. The users of a contract have even less information in this regard. Still users are expected to trust that the development team not only ensured their own security against attackers, but also ensured the users' security against the developers.

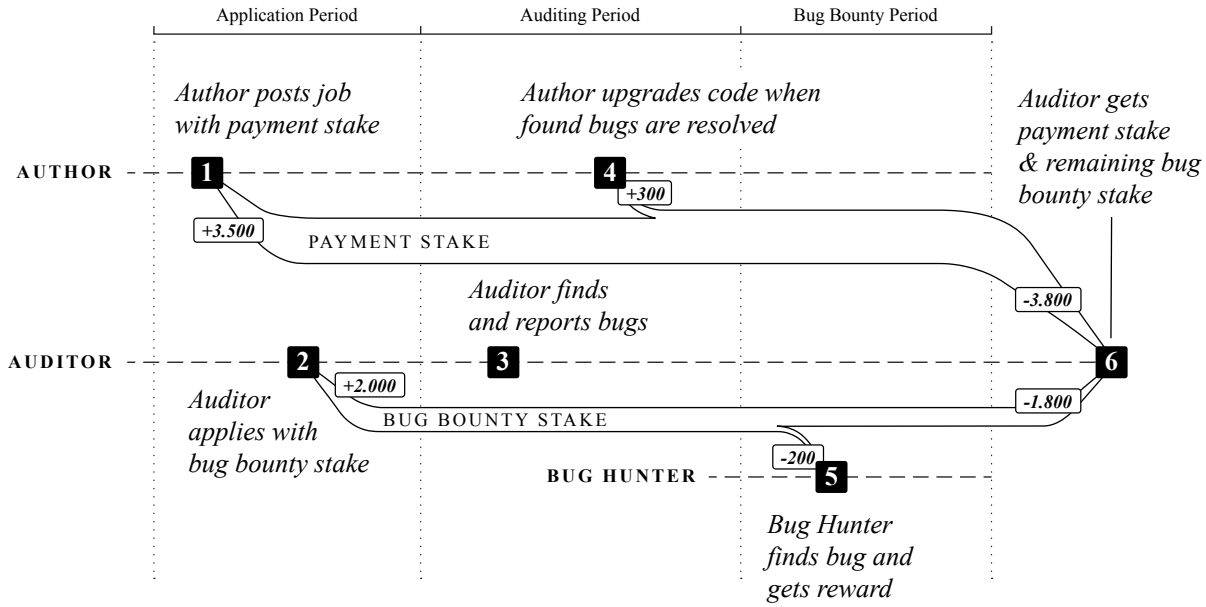
Ultimately, auditors risk only their reputation, which has proven an ineffective deterrent. Additionally, this contributes to a bottleneck in the market: the risk when hiring an auditor is so high, that developers only consider the prestigious firms as options, which have very limited capacity, leading to a vicious cycle. We believe these problems can be solved; but first, a broad overview of the current audit process follows:

1. Once contract authors have finalized development, they seek an auditing firm, compare estimates, and select an auditor they trust;
2. The auditor is paid a previously agreed upon sum on delivery of an audit report consisting of a list of issues they have discovered in the contracts;
3. Contract authors update the code, resubmit the code to the auditor, and (sometimes) pay an additional fee to verify their fixes;
4. Authors repeat step 3 until the auditor approves the code;
5. Contract authors can optionally run a bug bounty program: a time-limited window preceding deployment, in which the public can submit bug reports, and if bug reports are confirmed the reporter is paid a reward proportional to the severity of the vulnerability;
6. Once the bug bounty program ends and the authors have fixed any issues that arose, the contract is deployed; and
7. A tiny fraction of projects run indefinite bug bounties.

We propose an alternative system for securing smart contracts, featuring a platform token called "Solid Tokens" (or SOLID) which are primarily used by auditors and bug hunters as a form of collateralized

reputation. Generally, these actors would stake Solid Tokens when making security assurances, lose their tokens when such assurances are incorrect, and gain tokens when they help secure contracts through audits and bug reports.

1.1 Audit and Bug Bounty



We expect the proposed process would work as follows:

1. A contract author has code that they believe is ready for audit. The author posts a job (code and intended behavior) with a payment stake (denominated in SOLID), auditing period, and bug bounty terms;
2. Auditors post bids with a bug bounty stake (denominated in SOLID); winning bids could be determined algorithmically or manually by the author. The winning auditor's stake would then be committed to the bug bounty pool and would be unlocked at the end of the bounty period. In short, the auditor commits to finding all bugs in the given codebase, else the payments for bug bounties in the bug bounty phase come out of this stake;
3. The auditor would have until the end of the auditing period to publicly file any issue(s) they discover;

4. In exchange for the auditor validating any fixes that were made in light of their report, an auditor and author can mutually decide to update their agreement with new code and/or spec. This could be done for an additional fee (significantly less than the initial audit cost) paid by the author and added to the payment stake;

5. During the bug bounty period, any user can file bugs against the code. If the issue is legitimate and not previously documented in the auditor's report, the bug hunter would receive a payment proportional to the severity of the bug from the auditor's bug bounty stake; and

6. If the bounty pool is depleted during the course of the bounty program, the auditor would forfeit the payment stake, and it would be returned to the author (who would be expected to then repeat the process with another auditor). Otherwise, the auditor would receive the entirety of the payment stake plus whatever remains of their bug bounty stake.

1.2 Bug Prediction Market

1. We assume, that after an author has fixed all previously found issues, the author would be ready to deploy;

2. The author would open a prediction market on the final code by providing initial liquidity (denominated in SOLID) to the automated market maker, and an end date. The funds provided for liquidity are used to fund the automated market maker, and are "payment" for the information that the prediction market provides.. The prediction market is planned to be tied to the bytecode of contracts, not a specific deployment. Authors would be able to open the bug prediction market on their code at any time, and we expect them to do so before deployment to gain final reassurance before going to mainnet;

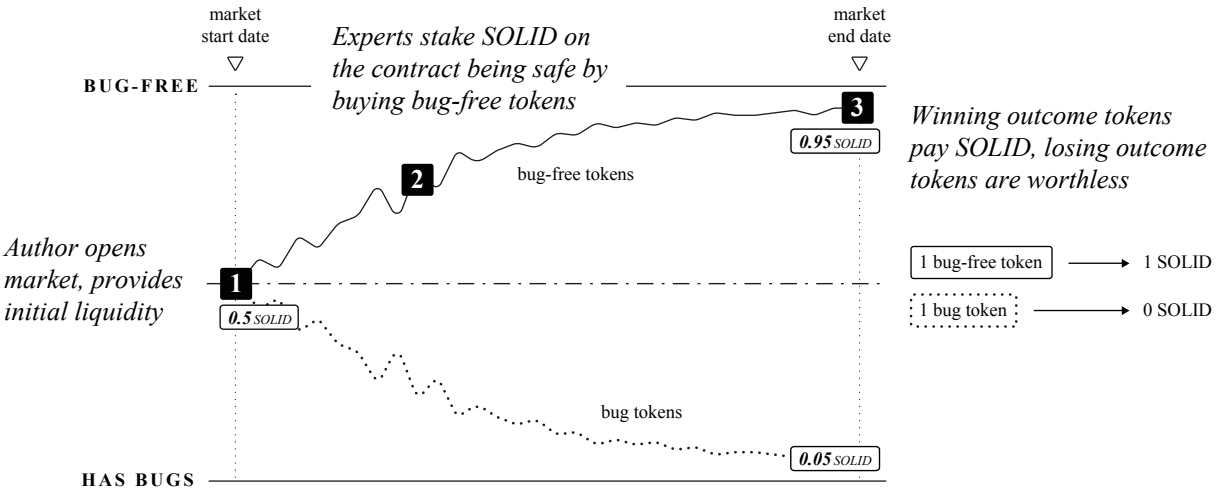
3. Up until the market end date, traders (themselves expected to be smart contract security experts) would be able to purchase or sell tokens representing the outcome (outcome tokens) of the market. If a trader believes a bug will be found in the code before the end date, they would be able to buy "bug tokens": we plan for such outcome tokens to pay out 1 SOLID in the event of a

bug being found. If a trader believes a bug will NOT be found in the code before the end date, they would be able to buy "bug-free tokens": we plan for such outcome tokens to pay out 1 SOLID in the event of no bug being confirmed by the end date. We expect the spot price of these markets to be the aggregate probability estimate of all traders in the market for the outcome. In other words, the spot price reflects the community's confidence in the security of the code;

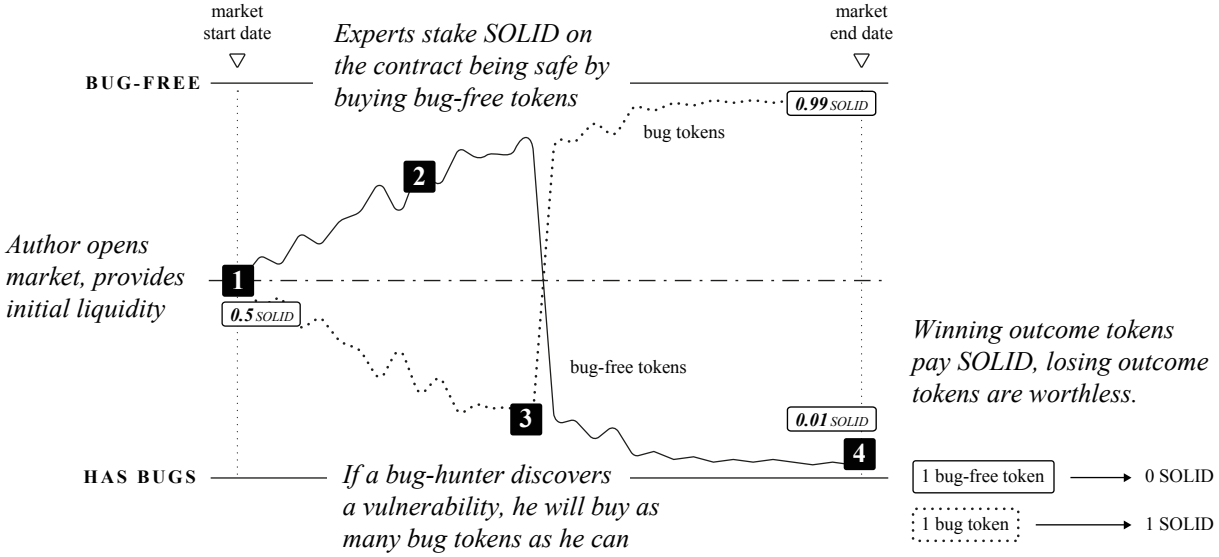
4. Third parties would be able to use the security confidence estimate to inform their decisions as to which smart contracts they interact with and how much value they trust them with;

5. Once the market's end date is reached, traders with outstanding tokens of the correct outcome would be able to redeem their outcome tokens for 1 Solid Token each. Tokens of the incorrect outcome (the event that didn't happen) are worth nothing. The end result would be that traders who predicted correctly are rewarded; and

6. After the initial market's expiration, if an author wishes to continue displaying their community security confidence metric, such author would choose to create a new market on the same code.



Scenario 1 — the code appears to be bug-free, thus the price of bug-free tokens stays close to one. At the same time, the value of bug tokens stays close to zero respectively.



Scenario 2 — a bug hunter discovers a vulnerability in a smart contract that was already considered as very secure – by buying as many bug tokens as possible, he maximizes his reward.

2 Motivations

Our main motivation behind the proposed system is to allow smart contract developers (and possibly smart contract users or other stakeholders) to transfer the burden of auditing and certifying the security of their contracts to a highly competent decentralized market. Auditors currently take on comparatively little risk when producing an audit. We believe that as a result, auditors incentives aren't strongly aligned with the goal of finding all the bugs in a set of smart contracts. We believe the auditor should be staking an economic representation of their reputation, otherwise the network will likely be insufficiently resistant to sybil attacks and exit scams. We plan to make any successful bug report during the bug bounty unfavorable to the auditor in order to disincentivize the auditor from posing as a bug hunter and posting bugs inside the bug bounty, instead of identifying them during the audit process. To this end, the auditors would bid platform-specific tokens representing qualified work, called Solid Tokens (or SOLID), to attempt to win audit jobs denominated in these tokens. Auditors would need to stake, at a minimum, the amount of tokens they believe would be necessary to catch any issues they might miss, or would risk forfeiting their payment.

In this system, every endorsement is economically backed by auditors. Auditors who make incorrect assessments would have their stakes transferred to those who correct them. The risk involved in securing smart contracts would be accurately modeled and capitalized, instead of externalized. We expect contract authors will be reassured that experts vouching for them have sufficient skin in the game. Currently, contract authors typically pay tens of thousands of euros for uninsured audits (audits costing hundreds of thousands of euros are not unheard of), then risk similar amounts of money in the form of a bug bounty without a strong signal of the actual risk. We believe our proposal would provide transparency and accountability to this highly valuable process.

3 Proposed Platform Process

3.1 Audit Request

Contract author posts initial request that contains:

- Smart contract code
- Falsifiable statements about the code that expresses its intended behavior (e.g. “The ETH cannot be stolen from this account. Buy function cannot be locked,” etc.)
- Payment stake in Solid Tokens (paid by the author)
- Auditing period (how long the auditor has to complete the job)
- Bug bounty terms (duration and reward tiers)
- Optional additional description of the code (e.g. known weakness, etc.)

Auditors would submit bids with a bug bounty stake: the Solid Tokens they personally risk during the bug bounty phase. If the author does not care about the soft reputation of the auditor (i.e. the auditor’s brand) an algorithmic auction could be held to determine the winner of the job (e.g. highest bid for bounty stake wins). Otherwise, it is up to the author to select from the competing bids.

3.2 Auditing Phase

Once the auction concludes, the auditor has until the audit period expires to attempt to secure the code in any way they see fit. The auditor can either audit the code on their own and file uncompensated bug reports (equivalent to an audit report), or they can transition to the bug bounty phase early.

3.3 Bug Bounty Phase

The bug bounty would be managed by the auditor, who approves bug submissions (though the reward sizes would have been previously decided in the auction phase). In the event of a disagreement between the bug reporter and the auditor about the validity of a submission, the bug reporter would be asked to post a stake to a separate anti-spam stake that is intended to cover the cost of evaluating the submissions by the auditor. This anti-spam stake would cover arbitration fees, and an equal amount would be staked from the tokens remaining in the reward pool. The size of the stake is planned to be determined by periodic voting by jurors. The stake of the winning party gets refunded, while the stake of the losing party is used to cover arbitration fees.

In addition to the arbitration stake, we estimate there could be an anti-spam stake, which would be refunded in the event the submission turns out to be valid.

For a bug report to be considered valid, we expect one basic requirement: that no bug report with equivalent meaning has been posted on the same code before, either by the auditor or other bug reporters. We expect this would be validated by the jurors who review the previous reports prior to accepting new reports. The same applies for bug reports posted during the prediction market period.

3.4 Bug Prediction Market Phase

We anticipate users will open a prediction market on whether a major vulnerability exists in a given set of smart contracts by providing initial liquidity (in SOLID) to the automated market maker. We would not expect traders to use a traditional order book for trading. Instead we plan

to use a hybrid model consisting of an automated market maker which serves as a "market maker of last resort" that is always available to trade against; in conjunction with batch auctions (discrete-time auctions in which all participants trade at a uniform price) which trigger when supplied sufficient liquidity[2]. We expect that when an order is filled, the clearing price will equal the market's estimation of the probability of the event occuring.

To further illustrate:

- If bug-free tokens have most recently traded at a price of 0.9 SOLID, the community confidence in the security of the given smart contract can be said to be 90%; conversely we can reasonably expect that bug tokens would be traded at a price of 0.1 SOLID, with an implied probability of 10% of a bug being found within the market window. In the event a bug is found (and confirmed through Solidified's arbitration process; see section 4.6) by the market end date, holders of that market's bug tokens can exchange their bug tokens for SOLID at a 1:1 rate via the prediction market smart contract. Holders of the bug-free tokens in this scenario are unable to exchange their tokens with the prediction market smart contract, i.e. the value these bug-free tokens is 0.
- A bug hunter who finds a bug in the subject of a given prediction market would likely acquire as many bug tokens as they can to maximize their reward: if the contract is widely considered secure, the reporter stands to be rewarded substantially as the bug tokens would be relatively cheap.

Bug reports could be encrypted and shown preferentially to the author to give a responsible disclosure window before initiating the bug confirmation process. We expect that the bug arbitration would be handled by the same validating system as the bug bounties.

Jurors could collect fees when the network determines they have voted truthfully in bug validity disputes. These fees would be covered by the bug reporter and/or holders of the bug tokens. Traders are expected to be smart contract security experts.

---

[2] http://cdetr.io/smart-markets/

This prediction market system aims to: (i) incentivize reporting bugs found in deployed contracts rather than exploiting them, (ii) provide an economic measure of the auditing community's confidence in the security of deployed contracts, (iii) potentially allow bug hunters to earn compensation for evaluating a bug-free smart contract, and (iv) serve as an early warning system for stakeholders of a smart contract when a vulnerability is discovered.

4 Mechanisms

4.1 Code Revisions

During the audit phase, the contract author may wish to move their (payment) stake to a revised version of their code that fixes found issues. Because the auditor may perceive such migration as riskier for themselves and they need to validate the fixes, the author can offer compensation for the migration.

This compensation would have to be negotiated separately for each migration, as there is no way to guarantee how much work a given revision demands. If an auditor were to refuse to migrate their stake, the author would have to find another auditor. We anticipate this would only happen in cases where the author is not offering fair compensation for a revision.

4.2 Collective Auditing

Because auditors tend to take on considerable risk when they accept an audit request and need considerable capital to do so, we believe it's likely that there would be demand for a system that spreads this task and the associated risks among multiple parties. For this purpose, special token pooling smart contracts could be created that would allow auditors to collectively source the bid on the audit request and form auditing collectives. We expect that this would be an appealing plugin for existing audit firms and professional auditor groups.

4.3 Bug Log Registry

The platform requires a single source of truth with respect to material necessary for bug report validity determination (i.e. the intended behavior, the source code, the compiled bytecode, and previously confirmed bugs). For this purpose, we plan to build a smart contract registry relating a hash of the contracts bytecode to the pertinent information hosted off-chain. IPFS is the most likely option for the needed off-chain data hosting[3]. However, since the availability of this data is crucial to the platform a further incentivisation scheme for compensating its hosts warrants further research. A fee for this registration could be charged as an anti-spam measure, and paid to the bug verification oracle (see section 4.6).

4.4 Certification of Compiled Bytecode

If there is a prediction market running on code with no known issues, we plan to issue certifications on bytecode that result from pairing the code with the current price of the bug free tokens on the market. We expect the price of these tokens would represent the platform's evaluation of the probability that the code is bug-free. This representation could be in the form of an embeddable badge, backed by a smart contract registry. Contract authors could use this in their marketing efforts: the goal is to provide an easily understandable security metric or stamp of approval for laymen, to whom smart contracts are otherwise totally opaque (even when the code is open source).

4.5 Prediction Market Participants' Incentives

We expect bug-free tokens will rise in price as the prediction market nears its end date. There are two reasons for this: (i) the contract has further proven its security and (ii) there's less time for a bug to be found. We expect that traders who get in early on secure contracts can take advantage from this upward trend. Importantly, we expect prediction market traders will have the opportunity to earn SOLID when the price of the tokens does not align with the true probability of the underlying

[3] https://ipfs.io/

event. Therefore, it is probable that this upward trend is counterbalanced: if the price rises above the actual probability of the contract being secure, traders can buy up bug tokens and potentially be rewarded. We expect purchasing and selling the tokens to move the price into alignment, and the underlying nature of the bug prediction market suggests price movement (i.e. opportunities for traders to earn SOLID) may be common.

Prediction markets are highly resistant to price manipulation.[4] For example, if a DApp developer wishes to make their own contracts seem more secure than they actually are, or a competitor's seem less secure, the developer could make large movements in the prediction market to this end. This developer is actually providing rational traders an opportunity to earn SOLID at great cost to themselves. Traders may take the opportunity and, in the process, will likely move the price back to rational consensus. One can imagine how continued price manipulation would likely be extremely costly in an efficient market.

4.6 Bug Verification Oracle

We plan for the validity of bug reports to be decided by a special class of users, called jurors. There are two problems related to incentivizing this class that we are trying to solve. First, since only a subset of jurors is selected to decide any given case (for more on this, see section 4.6.2), we believe that we need to incentivize jurors to adhere to the common standard of the whole juror community. It is presumed that there exists a sort of shared case evaluation method that is available to all jurors in the form of community culture, and we project following this method would, generally, lead to jurors picking the same outcome in specific cases, even in the absence of communication. This outcome forms what is known in game theory as a Schelling point.[5] In an attempt to incentivize individual jurors to adhere to the common standard and vote for outcomes that represent Schelling points, we believe we first need to reward jurors for picking identical outcomes and punish jurors who go against the majority vote. Second, the vote should be set up in such a way that it's near impossible to make any credible communication as

[4] http://bitcoinhivemind.com/papers/5_PM_Manipulation.pdf

[5] https://mindyourdecisions.com/blog/2008/04/01/focal-points-or-schelling-points-how-we-naturally-organize-in-games-of-coordination/

to what the juror's vote will be. If both of these conditions are met, we anticipate the optimal strategy of every juror would be to pick the clearest Schelling point, making the true outcome the ultimate result of the vote (for more on this, see section 4.6.3).

The second problem that we foresee when aiming to incentivize a class of jurors is attempting to ensure that jurors as a group would produce and maintain the type of voting culture that we believe leads to useful (honest and competent) results. It should be noted that if there is common agreement to defraud certain customers, a dishonest outcome may become a Schelling point. This outcome tells us that the Schelling game doesn't protect against malicious behavior of the juror community as a whole. Another useful point is that while the Schelling game should disincentivize subgroup coordination with respect to particular votes by punishing revelation of credible vote commitments, it isn't likely to prevent a majority stakeholder from privately communicating to other jurors that from now on he plans to vote in a dishonest way and that they should follow suit if they don't want to lose in coordination games.

So we expect that the foregoing shows that another incentive is needed, and such incentive is provided in the form of a juror's stake. Each juror would be required to stake a number of Solid Tokens, with the proportion of the stake in relation to the total stake of whole jury forming their probability of being selected to represent one vote in any given case. Should a juror want to stop working or want to reduce their stake, they would have to submit a request which deactivates the desired portion of the stake immediately, for the purpose of jury selection. The stake would become liquid, but only after a certain period of time elapses (yet to be determined). We expect this would incentivize the community of jurors to decide cases in a way that is useful to the client side, because arbitration fees are covered in the Solid Tokens and demand for Solid Tokens influences the token price and therefore value of a juror's stake. To reiterate, we expect behaving dishonestly to reduce the demand for arbitration, potentially damaging the value of the token before a malicious juror can liquidate their stake.

Even though our system should incentivize jurors to be loyal to the interests of the juror community and clients (in other words, it should guarantee honesty), it can't guarantee competent behavior. Discovery cost is individualized and higher than the cost of simply honest behavior (which is expected to be negligible). So, even in a community of honest jurors, there might arise a disagreement as to what is a compe-

tent judgement. We expect that it would become too costly to resolve this disagreement across the whole community. For example, if part of the community lacks the knowledge or reasoning ability to understand the argument of the dissenters, then there might arise a need to invite market forces to settle the dispute. For this purpose, we anticipate that our system would contain a mechanism for the community to split itself, where a portion of the Solid Tokens (and jurors' stakes) would be migrated to a different universe and would effectively become a separate currency. It should be noted that the need for this split doesn't have to be caused only by a difference competence, but also by a split in customer preference, where part of the customer community develops a distinct preference in judgement that differs from the rest of the customer community and creates a demand for a juror community with an equally distinct judgement culture. We can talk about "supply side initiated split" and "demand side initiated split", or "competence split" and "preference split". In the first case, a portion of jurors believes that they can better serve existing demand with different judgement culture. In the second case, a portion of the customers signals that they believe their needs would be better served by a different judgment culture and that they would increase their demand if a jury community with that culture were available.

4.6.1 Arbitration Fee

Jurors are planned to periodically vote on the minimum arbitration fee and jury size. Tentatively, the selected price would be the median price submitted, weighted by stake. Alternatively, the BVO could explicitly accept a limited amount of cases within a given time period, and auction these case slots. More sophisticated pricing schemes are being explored.

4.6.2 Jury Selection

The number of jurors selected is proportional to the fee paid. With each juror (one juror equals one staked Solid Token, not one expert) receiving an equal share of the fee. It is possible for an expert to be selected multiple times for the same dispute (e.g. one expert stakes three Solid Tokens, two of which are selected for a dispute), but mathematically, this is exceedingly rare given a sufficiently large juror population. Once a staked token is selected for a jury, it cannot be selected for further

juries until the initial dispute is resolved. This is to prevent under-collateralization. If this were not the rule and a juror (staked token) were selected for two concurrent disputes, they could only be punished once (i.e. disproportionately), because they only have one token that can be confiscated. We plan that deactivated stakes (stakes in escrow period prior to being withdrawable) will also be omitted from jury selection.

4.6.3 Schelling Game (voting process)

Once jurors have been selected, each juror should receive the dispute particulars: the code, behavior specification, and bug report. There would be a deliberation period (tentatively three days), in which a decision must be submitted or the juror's stake is confiscated. Once a juror has reviewed the evidence they must vote: valid or invalid. Valid means that the juror believes the report is true: the bug exists. Invalid means that either the juror believes that the bug does not exist, or determining the validity is not possible (or too costly to be reasonably determined). For example, the report is nonsensical or written in a language not known to the juror. At the end of voting, when votes are revealed, jurors who voted with the majority split the stakes of the losing jurors (e.g. 2 jurors vote "valid", and 1 votes "invalid"; the jurors who voted "valid" each receive their staked Solid Token + 0.5 Solid Tokens, the juror who voted "invalid" loses the entirety of their stake). Jurors do not submit their vote in plain text. They would submit the hash of a secret, consisting of a random string appended to their vote. We believe attackers need to be able to make a credible commitment (prove beyond doubt that they voted a certain way) to be able to maliciously coordinate (collude with other attackers). Claims of voting a certain direction can and should be ignored as there is strong incentive to convince individuals to make the incorrect decision: correct voters would receive the incorrect voters' stakes. To make sure jurors do not reveal their commitment, any user (not just other jurors) can "steal" a juror's stake by submitting that juror's secret (the vote and random string) before the reveal period. This way anyone who proves to another party that they voted a certain direction, risks said party knocking them out of the vote for this reward. Jurors who do not reveal their own secret during the reveal period are considered to have not voted and are treated as such: their stake is confiscated.

A number of Ethereum projects are pursuing similar schemes rooted in the Schelling game concept, namely: Kleros (generalized, decentralized

arbitration protocol)[6] and Augur[7]. Though both include modifications tailored to their particular use cases, we are looking to these projects for additional validation of the token oriented Schelling game concept.

4.6.4 Alternative Tacit Coordination Schemes

Alternatives to the Schelling game are being explored, namely SVD-resolution algorithm (Truthcoin-inspired approach)[8], robust bayesian truth serum[9], peer prediction mechanisms[10] [11], and crowdsourced judgement elicitation mechanisms[12]. The Schelling game mechanism was designed with the assumption that determining the true answer requires negligible effort/cost, which may not hold for our use case. This potentially changes the game's equilibrium away from truth telling. For the purposes of forking, mechanisms based on batches of decisions may be preferred to those based on individual decisions. The reasons are (i) that a different trend in decisions would be a stronger basis for a market splitting fork than any one particular case and (ii) it could increase the cost for malicious voters to coordinate.[13] Additionally, the Schelling game mechanism implicitly assumes the decision in question has symmetric pay-off: meaning they do not have an external incentive to vote any particular way over another. This is clearly not the case for our application, since a juror could very well hold a position in the dispute be arbited and also has to consider how the ruling will affect the future value of the token. This could lead to a significantly increased risk of coordination failure.[14] Considering these incongruities, we may find that one of the alternative mechanisms fits our use case better than the Schelling game. We plan to evaluate the proposed mechanisms experimentally before an ultimate decision is made as to the exact mechanics of the Bug Verification Oracle (BVO).

---

[6] https://kleros.io/assets/whitepaper.pdf

[7] https://www.augur.net/whitepaper.pdf

[8] http://www.truthcoin.info/papers/truthcoin-whitepaper.pdf

[9] http://www-bcf.usc.edu/~shaddin/cs699fa17/docs/BTS_robust.pdf

[10] https://www.aaai.org/ocs/index.php/AAAI/AAAI17/paper/download/14675/13807

[11] https://arxiv.org/pdf/1612.00928.pdf

[12] http://www.arpitaghosh.com/papers/elicit_arxiv.pdf

[13] http://forum.bitcoinhivemind.com/index.php/topic,112.msg342.html#msg342

[14] http://econweb.ucsd.edu/~vcrawfor/CrawfordGneezyRottenstreichAER08.pdf

4.6.5 Solidified Ltd as Centralized BVO

We expect the decentralized BVO to be the aspect of the platform which requires the greatest further research and development effort. Both the decentralized audit platform and the bug prediction market require a reliably competent and truthful oracle for bug verification. To bootstrap the platform, Solidified Ltd plans to provide centralized bug dispute arbitration while the final BVO is being validated. A modular architecture is planned to ensure applications built on our centralized oracle can be seamlessly transitioned to the decentralized oracle. With respect to arbitration fees, we expect the centralized oracle to charge SOLID in a fashion that mimics the eventual fee scheme of the decentralized BVO. There are two major disadvantages to the centralized oracle: (i) it is a single point of failure for the platform and (ii) we expect its throughput to be significantly lower than the decentralized version. For these reasons, we plan to move to the decentralized scheme as soon as the correctness of the BVO is ensured.

4.7 Prediction Market Structure

We wish to abstract the trading of outcome tokens as much as possible for bug prediction market participants. We intend for prediction market traders to focus solely on the question at hand: "Is this code secure and how confident are you about that?", rather than attempting to trade strategically. To this end we plan to employ two primary market mechanisms: (i) automated market makers and (ii) batch auctions. It should also be noted that traders will also have the ability to create "complete sets" of outcome tokens (1 bug-free token + 1 bug token) at anytime in an open market, by providing 1 SOLID to the market contract. This can also be done in reverse as well: provide a complete set of outcome tokens to the contract and you'll receive 1 SOLID in return. Traders can do this because the total value of a complete set is guaranteed to equal 1 SOLID, seeing as only one of the outcomes will come to pass (remember winning outcome tokens are worth 1 SOLID, losing outcome tokens are worth 0). Our bug prediction market will be built on top of Gnosis' prediction market smart contracts, and we plan to make heavy use of their supplementary infrastructure as well.[15]

[15] https://github.com/gnosis/pm-contracts/tree/master/contracts

4.7.1 Automated Market Makers

Currently, we plan to use LMSR-based Hanson's Automated Market Maker.[16] This automated market maker is available to trade with at any time, however each individual purchase moves the price. It does not charge any fees or "vig", and is funded by the contract author's initial liquidity. This market maker requires setting an elasticity constant: the greater it is, the less "reactive" the price is and the more initial liquidity needs to provided. The cost of "moving the price" is determined solely by the current price, the target price, and the elasticity constant; it is not affected by the depth of the trades conducted. The elasticity constant could potentially be set at the platform level to ensure a minimum level of liquidity in created markets.

4.7.2 Batch Auctions

Currently, we plan to use simultaneous Dutch auctions, inspired by the Gnosis DutchX.[17] These auctions clear at a uniform price: everyone receives the same rate. We believe this will be desirable for both traders, as they can lock in more favorable prices than Hanson's AMM allows; and end users of the security confidence metric, as we believe this will reduce its volatility. These auctions have additional properties which lend themselves well to on-chain trading (e.g. they're front-running resistant).

4.8 Governance & Arbitration Norms

We believe that Solidified's value as a platform is predicated on the BVO's correctness. Mechanically, the BVO is simply a tacit coordination scheme. We anticipate that the mechanics would enforce making the choice that is believed to be widely believed by other jurors to be the truth. In other words, we assume that the mechanics create the right incentives to reach a Schelling point. What they don't guarantee is whether they arrived upon a Schelling point that matches the ground truth. In our view, for a juror to vote honestly and usefully, they must know

[16] HANSON'S AUTOMATED MARKET MAKER, Henry Berg and Todd A. Proebsting

[17] https://edcon.io/assets/ppt/5.4/5.4main/5.4am/Stefan%20George-Introduction%20to%20the%20Dutch%20Exchange.pdf

what other honest jurors believe to be the truth. So, for Solidified to function, the BVO's mechanics alone are not enough. We believe that the Solidified community (or at least participants in the BVO) needs to develop, communicate, and train governance norms (norms for group decision making) by which to determine the validity of a bug.

As Vlad Zamfir lays out in his EthCC talk on governance: for governance norms to hold, they must be legitimate, where legitimacy is the "common knowledge that a governance process will be used rather than abandoned or replaced".[18] We believe the legitimacy of the platform would initially be supported by Solidified's reputation for attracting the best auditors. Solidified could sponsor some formalizations of governance norms. This would be in the form of a published guide and case studies on what constitutes concepts such as "a bug" and "severity".

As the platform grows, however, we anticipate that the single greatest source of legitimacy and norms would likely be the BVO's rulings on past bugs. We expect that the decisions the BVO has made in the past will form a system analogous to that of common law/case law. The coordination game underlying the BVO rewards truth only because truth is the Schelling point with the lowest discovery cost. When considering a bug that is analogous to a bug on which a ruling already exists, we project that the focal point with the lowest discovery cost is the existing ruling (in short, participants had already decided what the truth was previously and are expected to repeat that judgement).[19] For these reasons, we expect that it would be important for a single decentralized and sybil resistant discussion forum specific to bug rulings (current and present) to develop early legitimacy as the source of norms.

4.9 Forking

The goal of allowing forking is two-fold: (i) to make the BVO more robust against attacks that exploit core assumptions of its objective mechanisms (e.g. 51% attack and the P + epsilon attack)[20] and (ii) to potentially allow the market to self-select into separate niches. The exact nature of the forking mechanism depends heavily on the final coordi-

[18] https://www.youtube.com/watch?v=w8DjFbCTjus

[19] https://www.princeton.edu/~harman/Courses/PHI534-2012-13/Nov26/lewis-convention1.pdf

[20] https://blog.ethereum.org/2015/02/14/subjectivity-exploitability-tradeoff/

nation mechanism. We aim to find a balance between a purely mechanically enforced fork (i.e. token automatically splits into two mutually exclusive token contracts when a quantitative disagreement threshold is reached in the BVO) and a purely social one (i.e. a coalition of token holders migrating to a new token contract with altered balances). To this end we are closely following developments in the concept of subjectivocracy: a governance process by which systems can be allowed to split into multiple forks, and users opt into the fork they prefer.[21]

5 Solid Token

To reiterate, we intend for Solid Token to be used to (i) purchase security audits, (ii) place bids for audit jobs, (iii) back security assertions, (iv) fund and participate in bug bounties, (v) open and trade in bug prediction markets, and (vi) participate as a juror in the bug verification oracle.

Solid Token is planned to be an ERC20 token with a maximum total supply of 4,000,000, and no inflationary mechanism. Token burning would be used in the event of a fork. A fork results in the destruction of the original token and tokens can only be migrated to one of the competing forks. This would permanently decrease the token supply (more accurately, split it), but is not expected to occur often.

SOLID could be considered to have aspects of a medium-of-exchange token. Some observers are concerned that medium-of-exchange tokens will become worthless in the long run without consistent "sinks"—places tokens permanently disappear.[22] The argument is that if tokens are not required to be held for the protocol's mechanisms to function, then coins won't be held but rather purchased only when in immediate need and sold immediately after receiving: so as not to expose the actor to the volatility of the platform token. We believe that would not be the case with SOLID as nearly all of its mechanisms require staking and locking tokens, often for a period of many weeks. Users cannot participate in the system without holding (not just purchasing) the token in one way or another. We believe this would decrease the circulating token supply sufficiently to disincentivize the runaway velocity threat.

[21] https://github.com/realitykeys/subjectivocracy/blob/master/whitepaper.md

[22] https://vitalik.ca/general/2017/10/17/moe.html

Acknowledgements

Thanks to Todd Proebsting for his critique and insight into the practical considerations of running prediction markets & automated market makers; and Stefan George for bringing the concept of batch auctions in the context of prediction markets to our attention.