A Call for a Temporary Moratorium on "The DAO"

DRAFT (v0.3.2)

Dino Mark, Vlad Zamfir, Emin Gün Sirer
dino at smartwallet dot org, vlad@ethereum.org, egs@cs.cornell.edu
May 26, 2016
(revised May 30, 2016)

Over the past 3 weeks a Distributed Autonomous Organization (DAO) known simply as 'The DAO' and implemented as a smart contract on the Ethereum blockchain, has raised 11.5 million Ether, valued at $150 million at the time of writing. This is the largest crowd-funding event in history. The DAO now controls 16% of the total supply of Ether. It is arguably the most visible project in the Ethereum ecosystem.

In this paper, we analyze the rules of The DAO and identify problems with its mechanism design that incentivize investors to behave strategically; that is, at odds with truthfully voting to reveal their preferences. We then outline potential attacks against The DAO made possible by these behaviors.

In particular, we identify nine causes for concern that can lead DAO participants to engage in strategic rather than honest behaviors. Some of these behaviors can cause honest DAO investors to have their investments hijacked or committed to proposals against their interest and intent.

We discuss these attacks, and provide concrete and simple suggestions that will mitigate the attacks, or in some cases make them completely impossible.

We would like to call for a moratorium on proposals to prevent losses to the DAO caused by unintended consequences of its mechanism design. A moratorium would give The DAO time to make security upgrades, and should be lifted only once the DAO is updated.

Introduction

Smart contracts enable the collection and disbursement of funds according to immutable computer programs. Built on a Turing-complete platform, such contracts have the capacity to create constrained and predictable financial constructs without a trusted entity. Distributed autonomous organizations are one such class of contracts that can carry out corporate functions in accordance with the will of their shareholders, while being constrained by programmatic bylaws. These programmatic bylaws, if written with sufficient care can obviate the need for a management team in certain constrained domains.

Perhaps one of the most suitable such domains is crowd-funding. In traditional crowd-funding, a corporation such as Kickstarter connects investors with individuals or organizations who propose ventures. When the proposal gathers sufficient opt-in from the investors, it can proceed. If it succeeds, it returns financial rewards to its investors. The crowd-funding platform extracts some overhead for the matchmaking service it provides in the middle.

Another potential domain is investment funds or venture capital firms. In traditional venture capital firms, the managers collect funds from investors, evaluate proposals for various ventures, and determine a subset of ventures to fund. Successful ventures bring returns to the fund, from which the fund managers extract some, often substantial, overhead for the decision-making service they provide in the middle.

Over the last month, we witnessed the emergence of a distributed autonomous organization, known as The DAO, that is a cross between these two domains and seeks to completely eliminate the middlemen. The DAO operates somewhat like a venture capital firm, in that it collects a pool of funds to invest in worthy proposals, but it differs in that all the individual investors are able to vote, in proportion to the size of their investment, on each investment proposal put forward to the fund. The aspirational goals for The DAO are to utilize the wisdom of the crowds for this decision-making process, and to eliminate the risks posed by middlemen using a programmatic approach to corporate management.

The DAO is unique in many ways. It was funded through a crowd-funding effort that quickly raised 11.6M Ether (worth approximately $150M at the time of writing), making it the largest crowd-funded project in history. At this funding level, The DAO commands approximately 15% of the total Ether in existence. Because The DAO is so large, and because it is one of the first smart contracts of its kind, it has garnered much attention. Consequently, public opinion about decentralized autonomous organizations rides on its success.

Yet smart contracts pose unique technical challenges. Recall that computer programs can and most often do contain bugs. When a desktop application has a bug, it may crash; when a smart contract has a bug, it may render funds irrecoverable. Moreover, the smart contract cannot be easily updated, unlike desktop apps and other traditional software. Thus, careful thought and considerations must be put into constructing a smart contract that carries out the intended operations of a complex decision-making investment fund, especially in the presence of potentially malicious participants.

In this paper, we focus specifically on The DAO and examine the operational details of The DAO's smart contract with an emphasis on its mechanism design. We then identify nine causes for concern, where the mechanisms encoded into the current implementation of The DAO can give rise to unwanted strategic behaviors for the participants that are at odds with the primary function of the organization. In the case of The DAO, we show that in the current implementation can attacks with severe consequences are possible. We identify an attack that can indefinitely tie up investor funds and lead to ransom demands; an attack that enables a large cartel to usurp funds; and another attack that can enable an attacker to depress the value of the native fund tokens, among others.

At a fundamental level, these attacks all stem from unintended consequences of the mechanisms built into The DAO. Some are facilitated by an inherent bias towards voting to fund proposals; the current system discourages people from voting when they perceive a proposal to have negative expected value. A second fundamental problem stems from the structure of the withdrawal process: investors wanting to exit from the fund by "splitting" are vulnerable to attack. Combined, these problems can give rise to complex strategic behaviors, all resulting in a corruption of the intended, honest debate and voting process to select the most deserving proposals.

In the rest of this paper, we describe the operation of The DAO, the voting bias, potential attacks, and then discuss some potential mitigations and solutions. The central take-away from our analysis and discussion is that it would be prudent to call for a temporary moratorium on whitelisting proposals so that reasonable measures can be taken to improve the mechanisms of The DAO. Therefore, we call on the curators to put a moratorium in effect.

There are two alternatives to a curator-imposed moratorium. One is to ask The DAO token holders to place a self-imposed moratorium by voting down every proposal with overwhelming majority. Due to the flaws involving negative votes outlined in this paper, it would be a mistake to depend on this mechanism to protect against attacks targeting the same mechanism. The second alternative is to ask the DAO token holders to opt-in to the security measures by holding a vote for a new curator set who will implement a moratorium. We believe that The DAO's default behavior should favor security. Since no one knows the percentage of non-voting, non-active token holders, the threshold required for curator changes may be too high for the voting process to meet. For these reasons, we believe that the safest and most immediate course of action would be for the curators impose a moratorium, and allow the DAO token holders opt-out by means of a curator change vote.

The Structure of The DAO

The primary function of The DAO is to serve as a crowd-funding investment vehicle. To this end, The DAO API is structured around an initial creation phase that collects funds and an operational phase which consists of collecting proposals, voting on them, optionally funding them, and performing administrative functions such as paying out rewards and withdrawing funds. In the following discussion, we cover the operation of the DAO in each of these phases, and discuss the main abstractions behind the DAO to provide a context for game-theoretic analysis of the operation of this smart contract.

The DAO was created on April 30, 2016 at 10:00 UTC, based on a specific instantiation[1] of The DAO contract[2]. This paper describes the operation of this smart contract.

Creation and Funding Phase

The DAO creation phase started with the initial creation of the smart contract and lasted for 27 days. During this period, The DAO issues tokens, called The DAO Tokens (TDT), in exchange for Ether sent to a designated funding address[3].

The buy-in price of TDT varies during the creation phase. First, it starts at 1.00 ether for 100 TDT for the first 14 days. Then, there is an increase of 0.05 ether per 100 TDT for the following 10 days, then a final 3 day period at 1.50 ether per 100 TDT.

Late investors who paid more than 1.00 ether per 100 TDT have their surplus ether above 1.00 placed in a special account called extraBalance. Individual token holders cannot withdraw their funds from the extraBalance account; this money can only be moved after an amount equal to the extraBalance has been spent on proposals. In effect, the extraBalance represents additional money made available to the fund for spending on proposals, money earned by the DAO through additional fees paid by late joiners. For example, if a token holder paid 1.05 Ether for 100 TDT, and if no Ether had been committed to any proposals, the token holder could still only withdraw 1.00 Ether. The extra 0.05 Ether will stay locked in until The DAO has funded proposals that, in aggregate, exceed the amount of the extraBalance. Only then is the extraBalance folded into the main balance of the DAO, where it is distributed proportionally to TDT holders. At the time of writing the extraBalance is approximately 275,000 Ether.

The DAO follows a pattern[4] where the main contract acts as a factory for sub-contracts that split off from the main DAO. In what follows, we will refer to the initial contract simply as The DAO, its children as

---

[1] https://etherscan.io/tx/0xe9ebfecc2fa10100db51a4408d18193b3ac504584b51a4e55bdef1318f0a30f9
[2] https://github.com/slockit/DAO/blob/982e3c242ee31dfed4c04db79ea9751ac3e98efb/DAO.sol
[3] The ether address for The DAO is 0xbb9bc244d798123fde783fcc1c72d3bb8c189413
[4] Vitalik Buterin, Bootstrapping A Decentralized Autonomous Corporation: Part I. https://bitcoinmagazine.com/articles/bootstrapping-a-decentralized-autonomous-corporation-part-i-1379644274

child-DAOs, and collectively to any contract that implements the 'Standard DAO Framework' as a DAO[5]. The process of generating child-DAOs can continue recursively, until a depth limit is reached.

The Curator

Every instance of The DAO has a designated curator that is responsible for adding addresses to and removing addresses from the proposal payment address whitelist. The 'Curator' account for the current instance of The DAO is a 5 out of 11 multi-signature address (note that one of the curators has announced that they will not participate, although his key technically still has the right to sign in the multisig).

Only addresses on the whitelist can submit proposals to, and be funded, by The DAO. Proposals that want funding from the DAO must to ask the curator to add their address to the whitelist. Thus, the curator ensures that some human supervision is involved in the selection of proposals to be funded for the DAO. In effort to shield curators from legal liability, their responsibilities are limited strictly to deterring "malicious proposals." The main motivation for the curator abstraction[6] is a majority takeover attack where a large (53%) voting bloc votes to commit 100% of The DAO's funds to a proposal that benefits solely that bloc. The curator concept was introduced mainly to weed out such proposals and either refuse to whitelist their payment addresses or to un-whitelist their addresses; curators are expected not to take profitability or business sense into account while making whitelisting decisions. The task of exercising business judgment over the proposals is left up to the wisdom of the crowds through the proposal and voting process.

Proposals and Voting

Once a proposal has its address whitelisted by the curator, token holders can then vote on whether or not they want to fund that proposal. All TDT holders are allowed to vote either YES or NO, and their votes are weighted by the amount of their TDT holdings. The voting commences for a minimum voting period of 14 days, at the end of which the weighted votes are tallied. A simple majority of YES votes is required for a proposal to be successfully funded, and a minimum quorum of voters is required in order for the voting phase to be closed. The minimum quorum varies between 20 to 53% depending on the size of the proposal. Very large proposals will require a 53% quorum, while small ones only need 20%. There is no limit to how many proposals can be simultaneously going through the voting process. In order to prevent proposal spam, there is a non-refundable listing fee for each proposal.

Voting is an activity that limits future actions available to a TDT holder. Critically, if a token holder votes either yes or no on a proposal, they cannot change their vote, nor can they withdraw from the DAO through a split until the voting period has ended, nor can they transfer their TDT. Voting on any proposal places a TDT holder on a list of 'blocked' addresses that cannot perform splits or transfers. For a TDT holder who votes on multiple proposals, the block remains in effect until the latest of the voting deadlines.

---

[5] Though the term DAO is more broad and can refer to any decentralized organization governed by a smart contract, in this paper, it is used solely to refer to Slock.it's specific implementation https://github.com/slockit/DAO.

[6] download.slock.it/public/DAO/WhitePaper.pdf

If the proposal on which a TDT holder voted succeeds, the holder can only withdraw their share of the Ether balance that is left after the winning proposal has been funded.

In contrast, token holders that do not vote can withdraw from the DAO by initiating a split. Splits take 7 days to fork off the funds; consequently, a split initiated by a user 7 days ahead of a proposal's voting deadline can operate without any risk that her funds will be spent on that proposal.

Splitting and Withdrawals

The DAO does not permit funds to be withdrawn as Ether directly. Instead, token holders can take their TDT out by a process known as a 'split', a process that takes 34 days in total to complete and involves creating a new DAO.

The split process begins by having a token holder initiate a special proposal with a new curator address and a funding amount of 0 ether. The voting period on a split proposal lasts a minimum of 7 days. The outcome of the vote on a split proposal is inconsequential, as the proposal cannot be executed. Instead, the presence of a split proposal whose voting period has ended confers the right to split from The DAO to the parties who voted YES on it. This takes place when these parties call a function called 'splitDAO' to move their funds from The DAO into a newly formed child-DAO contract. This provides a way to withdraw one's funds from The DAO; namely, individuals who wish to withdraw from The DAO initiate a new curator proposal, where they themselves are the new curator, wait for the voting period to expire, and then transfer their holdings to a newly created DAO.

When a token holder splits from The DAO through the above mechanism, the usual 27-day creation period for a new DAO still applies. This means that the whole process takes 34 days in total to initiate a split proposal (day 0), gather votes (for 7 days), split from The DAO, and then wait for the new DAO to be formed (for 27 days). The actual transfer takes place on the 7th day and the funds are tied down for 27 days.

When a token holder has successfully split into their own new DAO, they can create a proposal to pay themselves out the full balance of all the Ether left in the new DAO.

Transferability of TDT

TDTs that are not blocked due to voting are fully transferrable to any valid Ethereum address, and therefore can be sold immediately on exchanges or over the counter. Thus, if a token holder does not want to wait 34 days to split from The DAO and withdraw their ether, they can just sell their TDT tokens directly on exchanges for Ether, or perhaps even other cryptocurrencies such as Bitcoin.

Attacks and Concerns

Analyzing an investment vehicle such as The DAO is difficult. This is partly because game theoretic treatments typically require a full characterization of the actors, the potential moves available to them

within the game, and the various payoffs associated as a result of each move. In an interconnected financial system involving convertible assets with a large number of complex actors, there are many potential payoffs, not all of which can be expressed within the narrow confines of a game. That is, not all actors try to maximize their returns in Ether, and instead may have exogenous payoffs in dollar terms that are difficult to capture. For instance, an actor who has purchased put options on ether and damages the system's reputation via an attack on The DAO may well lose tokens in the game but come out ahead financially, and modeling their profit requires quantifying social factors and market effects. Many previous attempts to apply game theory to distributed systems or complex agent systems have suffered from simple-minded modeling that has, at times, led to incorrect conclusions. Consequently, we do not attempt to provide a full game theoretic treatment of The DAO in this paper. Instead, we discuss the guiding principles for good mechanism design that ought to apply to crowd-funding investment vehicles such as The DAO, and identify several weaknesses in the current structure of The DAO that violate these principles and open the shareholders to attack.

Guiding Principles

The central point of the DAO is to enable token holders to vote on proposals. A rational actor will cast her vote in a manner that is informed by the net present value she perceives for each proposal. Every proposal has a clear present cost, specified in the proposal itself. It returns value to the shareholders either through an expected profit denominated in ether and paid back to The DAO, or through the implicit appreciation of the TDTs. As with every investment, proposals to the DAO have a probability of success that depends on the nature of the venture and its business plan. For instance, a proposal may ask for 1000 Ether to make 1000 T-Shirts, and may estimate that they will sell 1000 T-Shirts at a profit of 5 Ether each over a time frame, and thus estimate they will return 5000 Ether to The DAO. It is expected that vigorous debate and discussion during the voting phase will enable each voter to independently estimate the chances of success, and thus, the expected value (EV).

A DAO is considered to have good mechanism design if actors incentivized to vote truthfully in accordance with their estimates of the expected value of each proposal. For the wisdom of the crowd to manifest itself, we would like a TDT holder to vote YES for a proposal that they believe has positive expected value (+EV), and NO for a proposal they believe has a negative expected value (-EV); alternatively, they may abstain if their vote is not going to change the outcome. We now describe why the current implementation of The DAO fails to uphold this principle.

The Affirmative Bias, and the Disincentive to Vote No

The current DAO has a strong positive bias to vote YES on proposals and to suppress NO votes as a side-effect of the way in which it restricts users' options after they vote. Specifically, the current DAO blocks token holders from splitting from the DAO or from selling their TDT once they have voted on a proposal, until the voting period ends. Thus, a voter who believes a proposal has a negative expected value is in a quandary: they can split from The DAO immediately without taking any risk, or else they can vote NO and hope that the proposal fails to be funded. A NO vote is therefore inherently risky for an investor who perceives the proposal to be -EV, in a way that voting YES is not for a +EV voter. As a

consequence, The DAO voting is likely to exhibit a bias: YES votes will arrive throughout the voting period, while a strategic token holder will want to cast their NO vote only when they have some assurance that the outcome of the vote will be NO. Strategic NO voters will cast their votes only after gaining information on others' negative perception of the same proposal, so the voting process itself will not yield reliably signal information about the token holders' preferences over the course of the voting period. Preferences of the positive voters will be visible early on, but the negative sentiment will be suppressed during the voting process -- this can result in an affirmative bias that can be a problem for a crowd-funding organization where YES results in funding projects.

The Stalking Attack

Splitting from The DAO (the only existing method of extracting one's Ether holdings from the main DAO contract) is currently open to a "stalking attack." Recall that a user who splits from The DAO initiates a new DAO contract in which they are initially the sole investor and curator. The intent is that a user can extract his funds by whitelisting a proposal to pay himself the entire contents of this contract, voting on it with 100% support, and then extracting the funds by executing the approved proposal.

However the split and the resulting sub-contract creation takes place on a public blockchain. Consequently, an attacker can pursue a targeted individual by buying tokens during the creation phase. Since a splitting user is the new curator of the nascent sub-contract, a stalker cannot actually steal funds; the victim can refuse to whitelist proposals by the stalker (though note that, due to potential for confusion and human error, the expected outcome from such attacks is still positive). If the stalker commits funds that correspond to 53% or more of the sub-contract, he can effectively block the victim from withdrawing their funds out of the contract back into ether. Subsequent attempts by the victim to split from the sub-contract (to create a sub-sub-contract) can be followed in the same manner, trapping the victim's funds and prohibiting conversion back to ether. The attacker places no funds at risk, because she can split from the child-DAO at any time before the depth limit is reached. This creates the possibility for ransom and blackmail.

An initial response to this problem[7] suggested some remedies for preventing and counterattacking during a stalker attack. These remedies not only require unusual technical sophistication and diligence on behalf of the token holders, but are technically insufficient to deter stalking. The suggested prevention technique is to never split with a proposal on which any other party has voted YES. This is insufficient because an attacker can programmatically vote YES on every split with a dust account to earn the option to split, and then transfer his funds before invoking split. Once a victim finds herself in a child-DAO, Slockit has suggested two counterattacks for the victim to try to fend off the attacker. Neither of them are resilient against an attacker that creates dust accounts to vote, then transfers the money to the dust account that voted for the grandchild-DAO that the victim decided to pursue[8]. A victim who splits from the child-DAO will forfeit the rewards from the original DAO, as these do not flow to grandchildren, leaving them to the attacker -- the stalking attack can thus pay dividends, literally, for the stalkers. Overall, none of the purported defense mechanisms are guaranteed to win (they rely on launching trap and ambush attacks

[7] https://github.com/slockit/DAO/wiki/Why-The-Stalker-attack-is-a-non-issue
[8] https://medium.com/@tobyai/thedao-no-safe-withdrawals-9dd568510d73#.um413xx7e

against the attacker), withdrawals are not a constant time (and gas) operation, and much more importantly, not only do these counterattacks require immense technical sophistication, they would severely impact the user experience and overall user satisfaction.

The Ambush Attack

In an ambush, a large investor takes advantage of the bias for DAO users to avoid voting NO by adding a large percent of YES votes at the last minute to fund a self-serving proposal. Recall that under the current DAO structure, a rational actor who believes a proposal is -EV is likely to refrain from voting, since doing so would restrict his ability to split his funds in the case that the proposal succeeds. This is especially true when the investor observes that sufficiently many NO votes already exist to reject the proposal. Consequently, even proposals that provide absurdly low returns to The DAO may garner NO votes that are barely sufficient to defeat them.

This kind of behavior opens the door to potential attack: A sufficiently large voting bloc can take advantage of this reticence by voting YES at the last possible moment to fund the proposal. Such attacks are very difficult to detect and defend against because they leave little to no time for The DAO token holders to withdraw their funds. Among the current DAO investors, there is already a whale who invested 888,888 Ether[9]. This investor currently commands 7.7% of all outstanding votes in The DAO. For a proposal that requires only a 20% quorum, this investor already has 77% of the required YES votes to pass the proposal, and just needs to conspire with 2.3% of the token holders, in return for paying the conspirators out from the stolen funds.

The Token Raid

In a token raid, a large investor stands to benefit by driving TDTs lower in value, either to profit from such price motion directly (e.g. via shorts or put options), or to purchase TDTs back in the open market in order to acquire a larger share of The DAO. A Token-Value attack is most successful if the attacker can (i) incentivize a large portion of token holders not to split, but instead sell their TDT directly on exchanges, and (ii) incentivize a large portion of the public not to purchase TDT on exchanges. An attacker can achieve (i) by implementing the stalker attack on anyone who splits and then making that attack public on social media. Worse, since the existence of the stalker attack is now well-known, the attacker need not attack any real entity, but can instead create fictitious entities who post stories of being stalked in order to sow panic among The DAO investors.

An attacker can achieve (ii) by creating a self-serving proposal widely understood to be -EV, waiting for the 6th day before voting ends, and then voting YES on it with a large block of votes. This action has the effect of discouraging rational market actors from buying TDT tokens because (a) if the attackers proposal succeeds they will lose their money, and (b) they don't have enough time to buy TDTs on an exchange and convert them back into Ether before the attackers proposal ends, thus eliminating any chance of risk-free arbitrage profits. The combined result of (i) and (ii) means that there will be net

[9] From address 0x04c973aff06f64b880524f16ae8c821928233ee5

selling pressure on TDT, leading to lower prices. The attacker can then buy up cheap TDT on exchanges for a risk free profit, because he is the only TDT buyer who has no risk if the attacking proposal actually manages to pass.

The extraBalance Attack

In the extraBalance Attack, an attacker tries to scare token holders into splitting from The DAO so that book value of TDT increases. The book value of TDT increases because token holders who split can not recover any extraBalance, so as more holders split, the extraBalance becomes a larger percentage of the total balance, thus increasing the book value of the TDT. This attack is made more severe by the fact that once an amount equal to the value of the extraBalance has been spent, a proposal can be created to send any amount of eth to extraBalance and the curator is not able to prevent this via the whitelist.

Currently the extraBalance is 275,000 Ether, which means the book value of TDT should be 1.02. If the Attacker can scare away half the token holders, the TDT will increase in value to 1.04. If the Attacker can scare away ~95% of the token holders, the book value of the remaining TDT will be roughly 2.00. In this attack, the Attacking Whale would do the opposite of the token-value attack by creating a self-serving proposal with a negative return and then immediately voting YES on it with a large voting block of TDT, thus scaring all the token holders, and then giving them 14 days until the end of the voting period so that they have more than enough time to safely split. In this scenario, splitting will be risk free (assuming that it is not coupled with a stalking attack), since voting NO could result in losses if the attackers end up having enough YES votes.

The Split Majority Takeover Attack

Even though the DAO white paper specifically identifies the majority takeover attack and introduces the concept of curators to deter it, it is not clear that the deterrence mechanism is sufficient. Recall that in the majority takeover attack outlined in the DAO whitepaper, a large voting bloc, of size 53% or more, votes to award 100% of the funds to a proposal that benefits solely that bloc. Curators are expected to detect such instances by tracking identities of the beneficiaries. Yet it is not clear how a curator can detect such an attack if the voting bloc, made up of a cartel of multiple entities, proposes not just a single proposal for 100% of the funds, but multiple different proposals. The constituents of the voting bloc can achieve their goal of emptying out the fund piecemeal. Fundamentally, this attack is indistinguishable "on the wire" from a number of investment opportunities that seem appealing to a majority. The key distinguishing factor here is the conflict of interest: the direct beneficiaries of the proposals are also token holders of The DAO.

Reward Dilution

Another potential attack[10] against token holders who split is for the remaining token holders of The DAO to dilute the dividends they pay out to token holders who split. They can carry out this attack by funding proposals that cycle the fund's coins, issuing new reward tokens that dilute the rewards that come in from earlier investments. This attack stems from the way reward accounting lumps maintenance costs, internal

[10] https://www.reddit.com/r/TheDao/comments/4ljzic/can_the_split_dao_reward_token_mechanism_be/

transfers and genuine investments into a single proposal abstraction. It requires curator participation to launch, but well-meaning curators can inadvertently launch it when reorganizing funds or when the fund fires underperforming contractors; that is, operations which take coins out and return them as rewards.

Risk-Free Voting

A token holder can vote on proposals without committing to fund them[11], which is an enabler for launching other attacks and executing strategic behavior. To do so, the token holder simply votes with his funds as usual, but then, when the voting period is over, calls 'unblockMe' and executes a split before the proposal is executed. This decouples the attacker's funds from any risk he might take with them while voting, and enables a large voter to force bad decisions on the remaining token holders as he exits. It is not without risk, as he may be unable to unblock and split in time, but it is nevertheless possible, as correct execution depends on timing assumptions.

The Concurrent Proposal Trap

The structure of The DAO can create undesirable dynamics in the presence of concurrent proposals. In particular, recall that a TDT holder who votes YES on a proposal is blocked from splitting or transferring until the end of the voting period on that proposal. This provides an attack amplification vector, where an attacker collects votes on a proposal with a long voting period, in effect trapping the voters' shares in The DAO. She can then issue an attacking proposal with a much shorter voting period. The attack, if successful, is guaranteed to impact the funds from the voters who were trapped. Trapped voters are forced to take active measures to defend their investments.

Independence Assumption

A critical implicit assumption in the discussion so far was that the proposals are independent. That is, their chances of success, and their returns, are not interlinked or dependent on each other. It is quite possible for simultaneous proposals to The DAO to be synergistic, or even antagonistic; for instance, a cluster of competing projects in the same space may affect each others' chances of success and thus, collective returns. Similarly, cooperating projects, if funded together, might create sufficient excitement to yield excess returns; evidence from social science indicates that social processes are driven by non-linear systems.

Yet the nature of voting on proposals in The DAO provide no way for investors to express complex, dependent preferences. For instance, an investor cannot indicate a conditional preference (e.g. "vote YES on this proposal if this other proposal is not funded or also funded"). In general, the construction of market mechanisms to elicit such preferences, and appropriate programmatic APIs for expressing them, requires a more detailed and nuanced contract. This does not constitute an attack vector, but it does indicate that we might see strategic voting behavior even in the absence of any ill will by participants.

---

[11] https://medium.com/@tobyai/thedao-no-safe-withdrawals-9dd568510d73#.um413xx7e

Potential Mitigations and Solutions

There exist partial and complete remedies to some of the attacks outlined above. Discussion of these solutions is ongoing. The mitigations and solutions require either technical changes to The DAO or a social agreement among The DAO's curators, or both.

Supporting Withdrawals

A function that any token holder can call to have an instant and direct withdrawal of their share of the DAO's Ether to regular addresses (and that would allow them to claim future rewards from proposals on which they already spent Ether) would make the Stalker attack impossible. It would also significantly mitigate the Token-Value attack.

Many token holders currently seem to believe that they can withdraw from The DAO at any time. Guaranteeing that this can happen, without having to resort to complex defense mechanisms, will ensure that the token holders' expectations are met.

Post-voting Grace Periods

Adding a grace period after the end of the voting periods, but before the proposals can be funded/executed, would give token holders time to move TDT or to split the DAO after seeing voting results but before their money is spent. Voting periods and grace periods would not be allowed to happen concurrently, because voting tokens must be locked until all of the voting periods for the proposals those tokens voted on.

The addition of a grace period definitively solves the voting bias by allowing token holders to vote "no" without forfeiting their right to sell or split in response to the outcome. It also gives the curators time to defend the DAO against ambush attacks by un-whitelisting payment addresses after seeing the voting results. It significantly mitigates the majority takeover attack and the ambush attack, by letting token holders withdraw after the vote passes.

Shorter Voting Periods

Shortening the voting period on a proposal so that the voting period only occurs in the last 1-2 days of a 14-day or longer debating period would shorten the time for which tokens are locked. This mitigates the Token-Value attack, and also reduces the propensity for voters to wait until the last minute to vote so that their TDT are not locked up.

Vote "No" and Withdraw on an Affirmative Decision

Having a special vote whose semantics are 'NO_AND_WITHDRAW_IF_VOTE_SUCEEEDS' allows token holders to signal that they will leave the DAO if a proposal passes. "NAW" votes publicly indicate

that the voter believes this proposal would cause damage to the value of the TDT and no longer wants to be part of The DAO if it succeeds.

Waiting for Quiet

A potential defense to deter ambush attacks is to extend the voting deadline in response to last minute changes in the direction of the vote. While last minute votes are to be expected in a fair voting system, mechanism biases that incentivize token holders to sit on the sidelines can be countered by extending the voting period and giving people time to observe the direction of the vote and to participate.

Commit/Reveal Voting

A generally applicable technique is to have the TDT holders first commit to their (blinded) votes, and then to remove the blinding in a revelation phase at the end of the voting period. This has the downside that the client voters now need to be stateful in order to remember their blinding factor. Further, they can share their blinding factors with others in order to reveal, and even prove, the disposition of their vote. Most importantly, blinding the votes diminishes the value of The DAO's voting process: by design, the votes can no longer act as a signal to other TDT holders about the holder's financial preferences. The preference discovery process will thus end up shifting out of the smart contract into exogeneous mechanisms, such as message boards and the like.

Vote Delegation

TDT holders who do not participate in the voting process reduce the security of the system. One can improve participation, and thus improve security, by enabling TDT holders to delegate their vote to proxies. This delegation feature necessitates significant modifications and sufficient complexity to render it unsuitable as a short-term fix.

Reward Accounting

The reward dilution attack can be stalled by a more accurate accounting of when each DAO split. Proposals can then pay dividends according to the number of reward tokens outstanding at the time they split.

Curator-enforced Proposal Independence

The independence assumption may be maintained voluntarily by the curators by ensuring that the proposals that are eligible for voting at any given time are indeed independent from each other.

Upgrading the DAO

The DAO (1.0) has a built-in upgrade mechanism called “newContract” that moves all the funds into a new DAO (1.1). While this mechanism is available, it might be prudent to save it for dire emergencies. A softer upgrade path might be to place a moratorium on new proposals, to create new DAOs, and then to provide proposals to shift funds from the 1.0 version to the 1.1 version (or versions).

Summary and Suggestions

This paper outlined the operation of The DAO contract, which currently holds a substantial portion of the Ether supply and has generated much excitement about decentralized autonomous organizations and smart contracts. It also identified nine causes for concern, which might cause The DAO voters to deviate from a truthful strategy. Some of these behaviors have the potential to lead to financial manipulation and even loss. It finally identified some potential mitigations and solutions to some of these biases and vulnerabilities.

Given the concerns outlined above, we believe it would be wise for the curators to not whitelist any proposals until the DAO is upgraded to mitigate the potential attacks described in this paper.

NOTE: THIS DOCUMENT DOES NOT CONSTITUTE FINANCIAL ADVICE. WE ARE NOT, AND WILL NOT BE HELD, RESPONSIBLE FOR YOUR FINANCIAL DECISIONS.

Acknowledgments

Many thanks to Rick Dudley, Christoph Jentzsch, Andrew Miller, Gustav Simonsson, and Alex Van de Sande for their comments and feedback on this draft. We are grateful to Toby Hoenisch, who pointed out the Risk-Free Voting technique, and Meher Roy, who discovered the token dilution attack and made it public.