Swirlds and Sybil Attacks

Leemon Baird
baird@swirlds.com
June 6, 2016

How does the Swirlds platform avoid *Sybil* attacks, where hordes of sock puppet accounts from one attacker can manipulate the system? The short answer is that it uses proof-of-stake internally, but is flexible externally. So it can operate as proof-of-stake or proof-of-work. It can operate as permissioned or not permissioned. It can operate in many other modes, as well. But internally, it is proof-of-stake. Section 1 gives an overview of what that means, and how it works. Section 2 describes one specific scenario: proof-of-stake for an open, non-permissioned system built on a cryptocurrency.

Section 1: How the Swirlds platform works

It is useful to compare the system to blockchain. There are four main components of the Swirlds system, each of which can be compared to something related to blockchain:

- **Platform** - the platform can be thought of as being like an operating system

- **App** - anyone can build apps on top of the platform. An app is like a particular implementation of blockchain, which controls things like whether it is proof-of-work or proof-of-stake, and whether it is permissioned or not. Although the consensus algorithm is actually built in to the platform, the app is allowed to set various parameters that control how it works. So it may be useful to imagine that each app is like a different implementation of a different kind of blockchain.

- **Swirld** - each shared world (swirld) is like a separate blockchain network. All the swirlds created by a given app will work in the same way, but they each have their own separate history and set of members.

- **Member** - a participant in a particular swirld (a member) is the equivalent of a miner in blockchain. Each member can create new transactions, and put them inside new events (which are like blocks).

The core consensus algorithm can be thought of as using forms of voting, in order to achieve consensus with guaranteed Byzantine agreement. The algorithm specifies various things that happen when certain fractions of the population vote a particular way, such as at least half, or at least two-thirds. The result is a mathematical guarantee that various attacks cannot succeed as long as less than one-third of the population is dishonest.

But what is the "population" and what does "one-third" mean? Internally, the platform defines a record of *voting stake* for each swirld. For a particular swirld, this means recording a number for each member, called the voting stake. Votes are weighted by voting stake. The mathematical guarantees apply when all the dishonest members together have a total voting stake that is less than one-third of the total voting stake of all members in that swirld.

So how does a member get voting stake? The app defines how this is done. The app developer can choose one of several obvious approaches, or create a new approach. The following are some of the obvious approaches a developer might choose, for which the platform helps provide support:

- **Proof of stake** - each member can associate themselves with one or more Bitcoin wallets they own, and their voting stake is set to the total balance of those wallets. Or do the same with some other cryptocurrency, even one defined by the swirld itself.

- **Proof of burn** - the same as proof of stake, but the member must actually prove that they destroyed the Bitcoin in question. In other words, there is a fee that must be paid to join the swirld, and the voting stake is proportional to the amount paid. And again, this could be implemented with any cryptocurrency with real value, not just Bitcoin.

- **Proof of work** - a member can earn voting stake by solving a computational puzzle. This is similar to Bitcoin, except the cost is incurred to earn voting stake, rather than to mine a block. If an app chose this approach, then members in its swirld would each need to keep mining in order to keep up with the others, and not lose their ability to keep the system safe. The app can also make the voting stake decay over time, to encourage continual work.

- **Permissioned** - each member gets a voting stake of exactly 1, but they are only allowed to become a member if they have permission. As with other permissioned systems, the permissioning could involve a vote by the humans involved, or a proof of membership in some existing organization, or something similar.

- **Hybrid** - the original founders of a swirld each start with an equal voting stake. This is like a permissioned system. From then on, anyone can join the swirld, if any existing member invites them, so membership can spread virally. Each member will split their own voting stake with all those they invite. In this way, a member can invite 1000 sock puppets to be members, but all 1001 of them together will still have the same total voting stake as the member had originally. So sock puppets will not help in launching a Sybil attack.

- **Trivial** - every member gets a voting stake of 1, anyone can invite as many sock puppets as they like, which also get a voting stake of 1 each, and a Sybil attack is simply not defended against.

The simplest nontrivial approach is the hybrid one. This is probably the best one for casual, low-value swirlds. A simple business collaboration or game might use this. It is convenient to the users, but still prevents a single disgruntled user from disrupting everyone. Or, if it’s just friends that trust each other, then even the trivial approach could be good enough.

The safest approach is the permissioned one. This might be used by a small group of banks that need to record a ledger of their actions. Only banks in the consortium can become members of the swirld, and each bank is allowed to participate as only a single member. Although one bank might not fully trust any particular bank in the group, it would probably trust that there would never be a full third of the group acting dishonestly.

The safest approach for a large group of strangers is the proof-of-burn. An attacker can still achieve a one-third fraction of the population. But if the entry fee is set high enough, then the cost of doing so can exceed the benefit of launching such an attack.

For a large enough group, the proof-of-stake could be sufficient, without the burn. This would work if there are many participants who own large amounts of the cryptocurrency, in roughly equal amounts, and it is not expected that someone disruptive would join who owns more than a third of them put together.

Of course, an app developer could choose to do something more complicated. The swirld could start with the hybrid model, then allow users to sell voting stake to each other. Or it might be permissioned, but with the permissioning done within the swirld itself, by having an actual election, with the humans talking with each other prior to voting. Or it might start with proof-of-burn, and automatically transition to proof-of-stake once the total value gets large enough. In all cases, the consensus is always decided by the platform, using the current record of voting stake, as managed by the app.

Section 2: An Example Scenario

Imagine a community of members running a “swirld” (a particular Swirlds network) for some specific purpose, such as a public ledger. It is proof-of-stake, where consensus voting is proportional to each member’s ownership of some amount of a cryptocurrency, which will be called StakeCoin for this example. The ledger swirld is open, not permissioned, so we cannot trust all the members. The ledger swirld uses proof-of-stake rather than proof-of-work, so it is low cost. The question to consider is whether it can be made secure.

The system will be secure if no attacker can obtain 1/3 of the total StakeCoin owned by all the participating members put together. The ledger swirld will continue to function as long as 2/3 of the StakeCoin is owned by members who participate and are honest.

One way for an attacker to gain control is for them to talk with various StakeCoin owners individually, and buy their StakeCoins. This is similar to cornering the market on a commodity, or trying to buy enough shares in a company for a hostile takeover. It is not only an attack on the ledger swirld that uses the StakeCoin. It is actually an attack on StakeCoin itself. If one person can gain a near-monopoly on a cryptocurrency, then they can manipulate its market value, and arrange to repeatedly sell high and buy low. This can be very profitable in the short term, and will ultimately undermine trust in the cryptocurrency, and perhaps lead to it being universally abandoned. This is unrelated to the technology used. If you can gain ownership of the majority of the BitCoins in the world, or the majority of the US dollars in the world, or the majority of the corn futures in the world, then you can profitably undermine the system.

How can such an attack be avoided? The attack is harder if the cryptocurrency is both valuable and widespread. If it is valuable, then it will cost a great deal to buy a large fraction of the StakeCoin money supply. And if it is widespread, with many different people owning StakeCoin, then attempts to corner the StakeCoin market will become visible early on, which will naturally raise the price of a StakeCoin, making it even harder to gain the rest.

A second attack is to obtain an amount of StakeCoin that is small compared to all the StakeCoin in the world, but large compared to the amount of StakeCoin owned by the members participating in the ledger swirld. This can be avoided if StakeCoin was specifically created for use in this particular swirld. In other words, the cryptocurrency and the ledger swirld might be created simultaneously, and each help provide value to the other.

But then there is a chicken and egg problem. The newly-created swirld needs a valuable cryptocurrency from the start, for security. But the newly-created cryptocurrency needs time to grow in value. How can this be achieved?

One approach is to start with a consortium of, say, 10 large, respected corporations or organizations that are the founders. Each is given a large amount of StakeCoins to start with, and the system is structured so that the money supply will not grow quickly, and will have some ultimate size limit. Each founder has an incentive to participate as a member in the ledger swirld and the StakeCoin swirld, where StakeCoin itself is a swirld running on a hashgraph with the Swirlds consensus algorithm. Because there is no proof-of-work, it is inexpensive to be a participating member running a node. The founders are trustworthy enough that it is unlikely that any large fraction of them will collude to undermine the system. Especially since that would destroy the value of the coins they hold and the ledger they are running.

But wait, isn't that just like a permissioned blockchain? Yes, initially. But that is only to get it started. Over time, other members can join the ledger swirld. And other people can buy StakeCoin, either directly from the founders, or on an exchange. The ledger could even incentivize members to participate by paying tiny amounts of StakeCoin for participating, to encourage more people to join. Over time, it could become much more distributed, with the stake eventually spreading out, so that it becomes difficult for anyone to corner the market, even if the founders colluded. At that point, the cryptocurrency will have real value, the ledger swirld will have real security, the system will be open without permissioning, and no one will have to pay the costs of wasted proof-of-work computations.