LITEPAPER V0.3

OCTOBER 2023

WHAT IS SARCOPHAGUS?

Sarcophagus is a blockchain-enabled, cryptographically secure, general-purpose digital dead man's switch. It is autonomous, censorship resistant and immutable. It relies on modern smart contracting networks like Ethereum as well as Arweave's permanent file storage network; files you place in a sarcophagus are only unlocked and available to your designated recipient in the event of you failing to attest to the contract within the time period specified during its creation.

Historically, dead man's switches have been used in heavy machinery to ensure the operator is paying attention and is present at their post. However, in the recent decades it has become increasingly obvious that a digital version of this tool is needed. Small strings of data are becoming more and more important in the lives of humans, and it is time to create a tool to help tie sovereign humans and their machines a little closer together.

Bitcoin gave us the base of blockchain tech, it proved that making computers do physical things to supplant human trust (or violence) is not only sustainable, but approaches anti-fragility. Ethereum gave us the ability to execute code on a 100% uptime, decentralized and immutable computer. Arweave gives us the ability to economically store and retrieve files from a separate, immutable and permanent file system. Only the combination of multiple groundbreaking innovations in the past 10 years has made Sarcophagus possible.

The conductor must be actively holding down the lever to keep the train in motion.
Release of lever will activate the breaks and bring the train to a stop.

A dead man's switch is a mechanism (digital or physical) that triggers when the user FAILS to perform an action. A standard switch will trigger when the user performs a positive action, a dead man's switch triggers when the user fails to perform a specified action within a set time period.

WHY SARCOPHAGUS?

FLEXIBILITY, CERTAINTY, AND ROBUSTNESS

The goal of this project is to make secret/sensitive data recovery and transfer easier and more intuitive. With the impact of digitization taking root in all facets of our lives, having a secure and safe way to share those secrets is critical. More importantly, it is imperative that these secrets are shared at the correct time. Sarcophagus empowers anyone to securely manage sharing of anything digital to a chosen recipient(s) under their own terms. Utilizing decentralized networks, Sarcophagus gives control back to the user and removes the middle man in most cases.

Launching revolutionary concepts with tools of the past to handle critical tasks such as password recovery, via a centralized server (ZK or otherwise) is counterproductive. Only by melding new technologies together can the productization of a properly decentralized and digital dead man's switch occur.

LEAVING AN IMMUTABLE LEGACY

A 'secret' in the context of this paper can be considered a small piece of data. This data could be a private key to a crypto wallet, a master password, or even a password manager database. In recent years, these secrets have become the primary target of major, organized hacking efforts. The simple fact is that these secrets are worth an enormous amount of money and power in aggregate.

While crypto keys and credentials are the most obvious use cases for a digital dead man's switch, there are potentially thousands of use cases. Any time there is a secret that a user would like to share only in the event of their removal from control, a sarcophagus can be useful.

Generally, dead man's switches incur the grim nature of finite human lifespan and cognitive capacity. However there are many use cases in which the user of the switch does not die or have anything bad happen to them at all. The user may have just transferred to another position within a high-security organization and needs a way to securely pass credentials to another employee.

Some of the initial use cases for Sarcophagus include:

- **Data-at-rest Security/Continuity**
- **Crypto Key Management**
- **Will and Trust**
- **Emergency Communications**
- **Political Activism & Journalism**
- **Password & Credential Recovery**
- **Time Capsule**

BUILT USING ARWEAVE AND EVM BLOCKCHAINS

Data integrity is critical in the operation of a dead man's switch. If the data stored by the user is not available indefinitely, as well as immutable, the security model is broken. Only recently with the rise of Arweave's permanent file storage network have applications been able to store data that will live on forever, in its original form, at its original discrete location.

To learn more about Arweave and how its endowment-based payment architecture works, visit https://www.arweave.org/

The other core technology that has enabled the creation of Sarcophagus is the proliferation of Ethereum and other EVM-based blockchains. While Ethereum has been around a few years longer than Arweave, it is the base layer of security that allows users to bridge trust away from traditional institutions like banks and law firms, and supplant that trust into mathematics and independently-verifiable strong cryptography.

Ethereum was the first fully programmable blockchain to gain traction after its launch in 2015. The ability for developers to create immutable smart contracts that trustlessly execute code in the EVM (Ethereum Virtual Machine) has opened the doors for endless possibilities in trustless computing.

Along with Ethereum, a host of other EVM-compatible chains have launched. Some of these chains are built on top of Ethereum (Layer 2 Chains) like Arbitrum and Optimism, and some are standalone Layer 1 chains like Avalanche.

The way that Sarcophagus has been archeteched allows for the application to be flexible in which networks it uses. While starting with Ethereum, other networks will be supported quickly after launch. This will allow the user to match their network choice with their security needs.

MULTI-NODE ARCHITECTURE AND SHAMIR'S SECRET SHARING SCHEME

Sarcophagus allows users to choose which node(s) they will employ to manage the release of their secret to their designated recipient in the case of their failure to attest to the contract within the user-specified timeframe. The application also allows the user to choose how many nodes to employ, as well as the number of nodes out of the total that are required to release the secret.

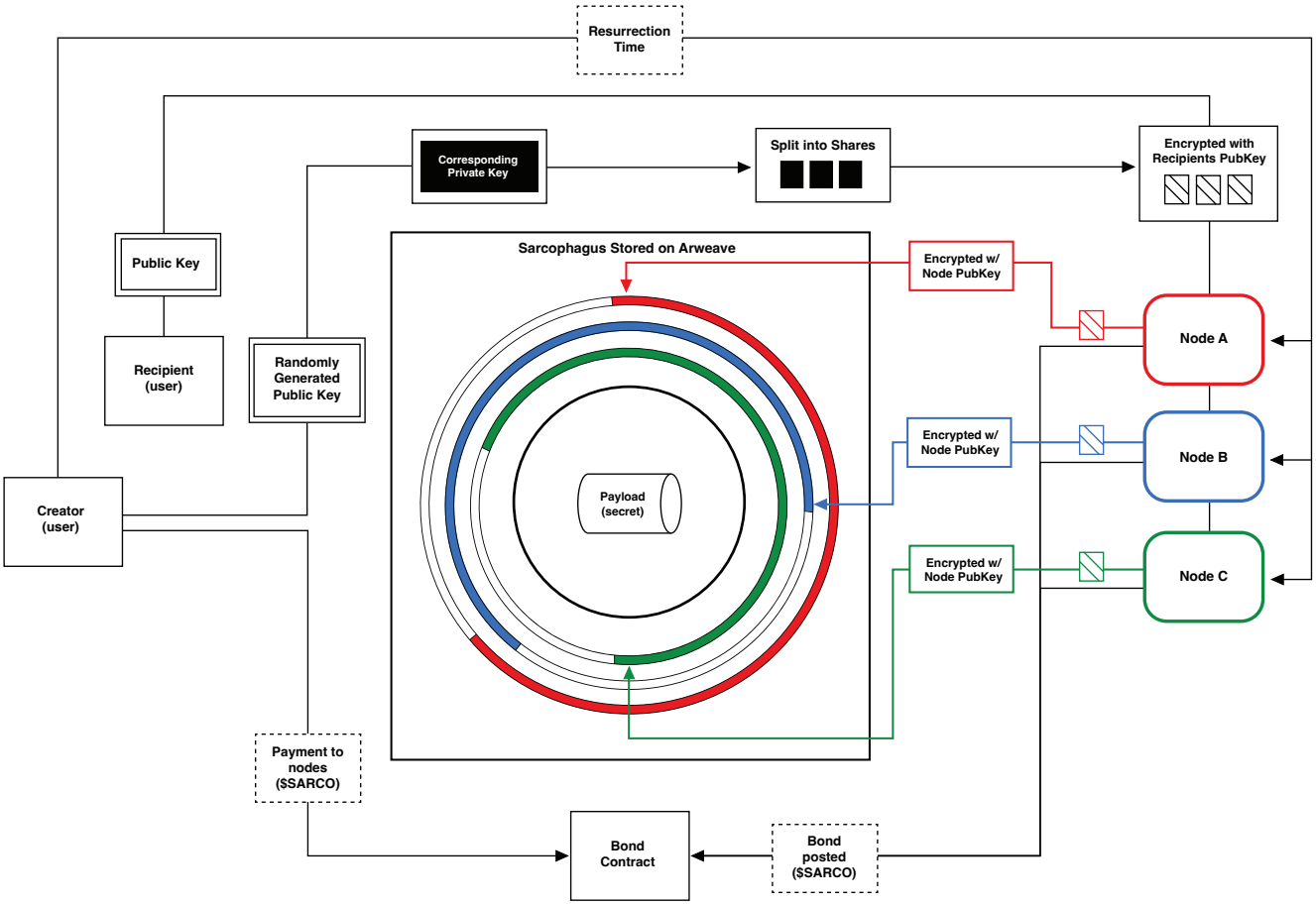
Based on Shamir's Secret Sharing Scheme, the application will display all available nodes, and allow the user to choose the number of nodes, along with the ratio of those nodes required to release their secret. For example: A user can choose to employ 3 nodes, but only require 2 out of those 3 nodes to participate in the release of the secret. This flexibility allows the user to create a more robust security model for each dead man's switch, while at the same time allowing for unforeseen circumstances where a single node may be offline when the secret is meant to be released. This is an extremely common scheme that is used in multisig contracts and currently protects billions of dollars in value throughout the crypto ecosystem.

CREATING A SARCOPHAGUS (DIGITAL DEAD MAN'S SWITCH)

In order to create a sarcophagus, the user must provide the application with the following inputs:

- The file containing the secret the user wishes to be released by the network if the switch is triggered. This is called the 'Payload'. It can be any type of file.
- The public key of the designated recipient. Only the user(s) controlling the private key of this address will be able to download the Payload in the event that the dead man's switch is triggered. The application also allows the user to generate a new ethereum keypair and download an easy-to-share PDF of the wallet. If the user wishes to make the payload visible to the public upon release, they would simply publish the private key of the recipient conspicuously.
- The discrete time/date in the future when the payload will be released if the user fails to attest to the contract. We call this the "Resurrection Time". This time can be updated at any point by the user prior to the triggering (by the nodes) of the dead man's switch.
- The total number of nodes the user wishes to employ to safeguard the secret. Each of these nodes will receive a share of the key required to trigger the dead man's switch.
- The number of nodes required to participate in the release of the payload in the case of the switch being triggered. If the user chooses to employ 3 nodes, and chooses to require 2 of the 3 nodes to participate in the release, 1 node could be offline at the time of triggering and the dead man's switch will still work perfectly. If the user chooses to require 3 of the 3 nodes to participate, it will be more secure, but less robust.
- Payments in ETH to fund file storage and $SARCO (ERC20) token fees to employ the nodes.

When the user chooses to employ a given node, fees in $SARCO are debited from the user, and simultaneously, $SARCO tokens are debited from the node. These tokens are placed in a contract that will hold them for the duration of the switch. If the node chooses to behave maliciously, the tokens locked in this contract will be slashed from the node and returned to the user. This mechanism provides an incentive for the node to continue to operate, and to operate in the fashion prescribed by the network.



I. 2OF3 SARCOPHAGUS CREATION

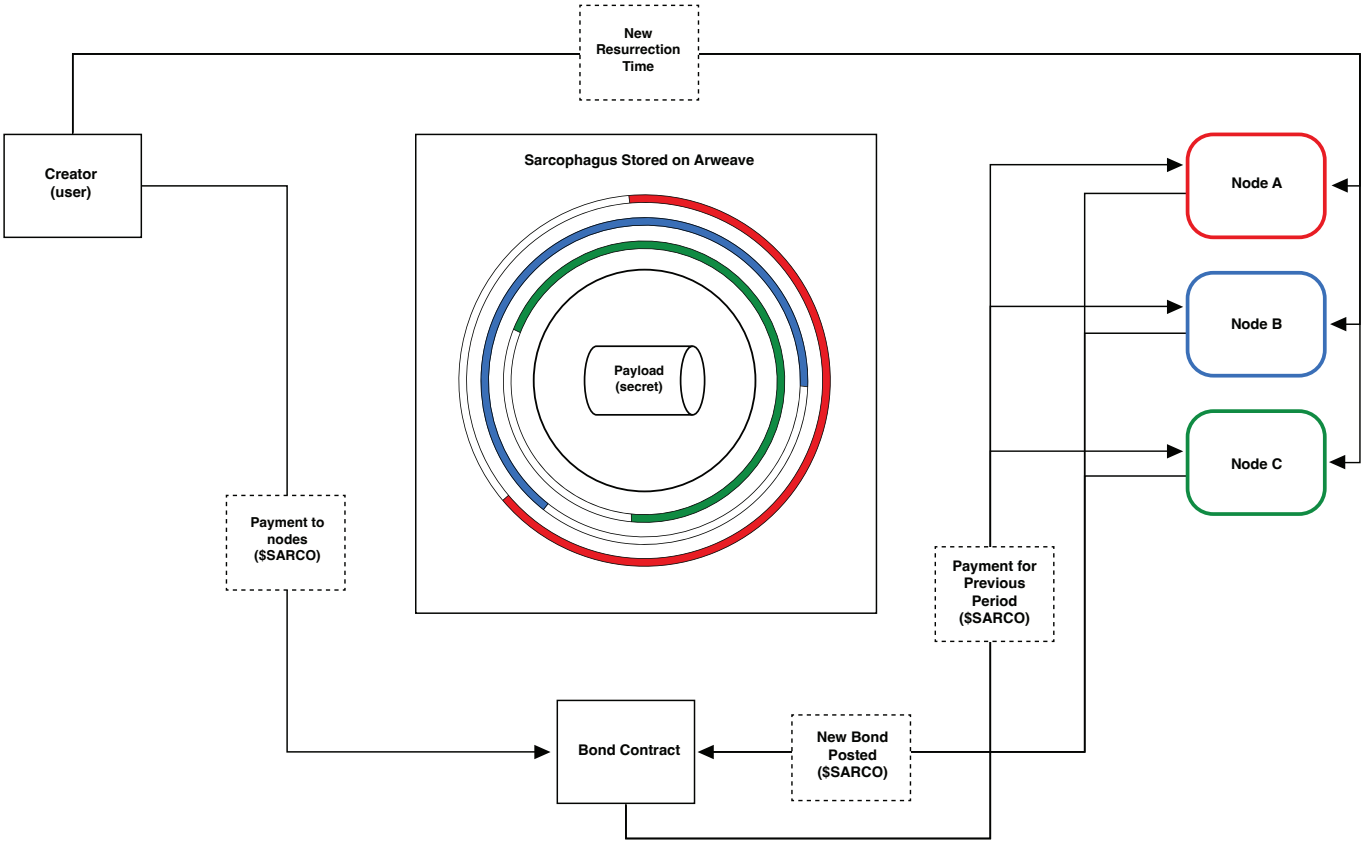
ATTESTATION AND ONGOING OPERATION

After the sarcophagus is created, the user must attest to the contract prior to the resurrection time to prove that they are still in control. This attestation can be seen as a proof of life, but it is a simple crypto transaction where the user proves control, and resets the switch to a new resurrection date. During this process the user must provide:

- A new discrete time/date in the future when the payload will be released if the user fails to attest again.
- Payments in $SARCO to the nodes for the next period of time.

When the user attests to the contract and chooses a new resurrection date, the $SARCO tokens that were held in the bond from the original sarcophagus creation are then transferred to the nodes as payment for services rendered, and a new round of fees and bonded tokens are placed into the contract. This process continues ad infinitum until the user fails to attest (read: is no longer in control) and the payload is released, or until the user chooses to bury the sarcophagus and end the relationship with the nodes they employed.

Since the fees charged by the nodes are time-based (e.g. 1 $SARCO/month), it is in the best interest of the user to choose a reasonable resurrection / release date each time they attest to the contract. There are no refunds paid to the user if they attest to their contract long before the payload is meant to be released.



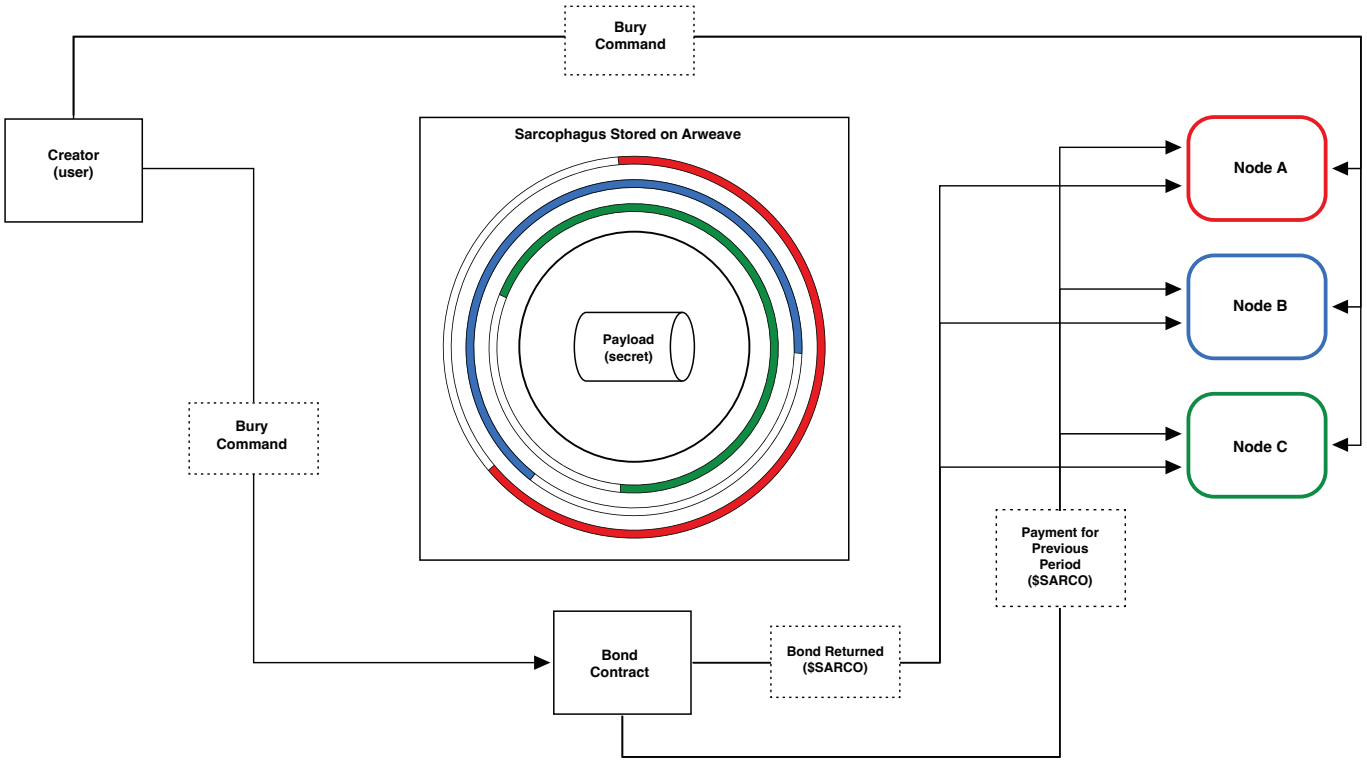
II. 2OF3 SARCOPHAGUS ATTESTATION

BURYING THE SARCOPHAGUS

If at any point the user chooses to discontinue use of the application, but does not want the payload to be released, they are able to 'bury' the sarcophagus and release the nodes from the contracted obligation they entered into.

When the user chooses to bury the sarcophagus they created; fees from the contract are paid to the node, and the node's locked $SARCO bond tokens are released back into their wallet.

The payload will continue to live on forever in its encrypted form on Arweave, but the nodes will no longer monitor the contract for attestation, and the keys will be purged from their system.



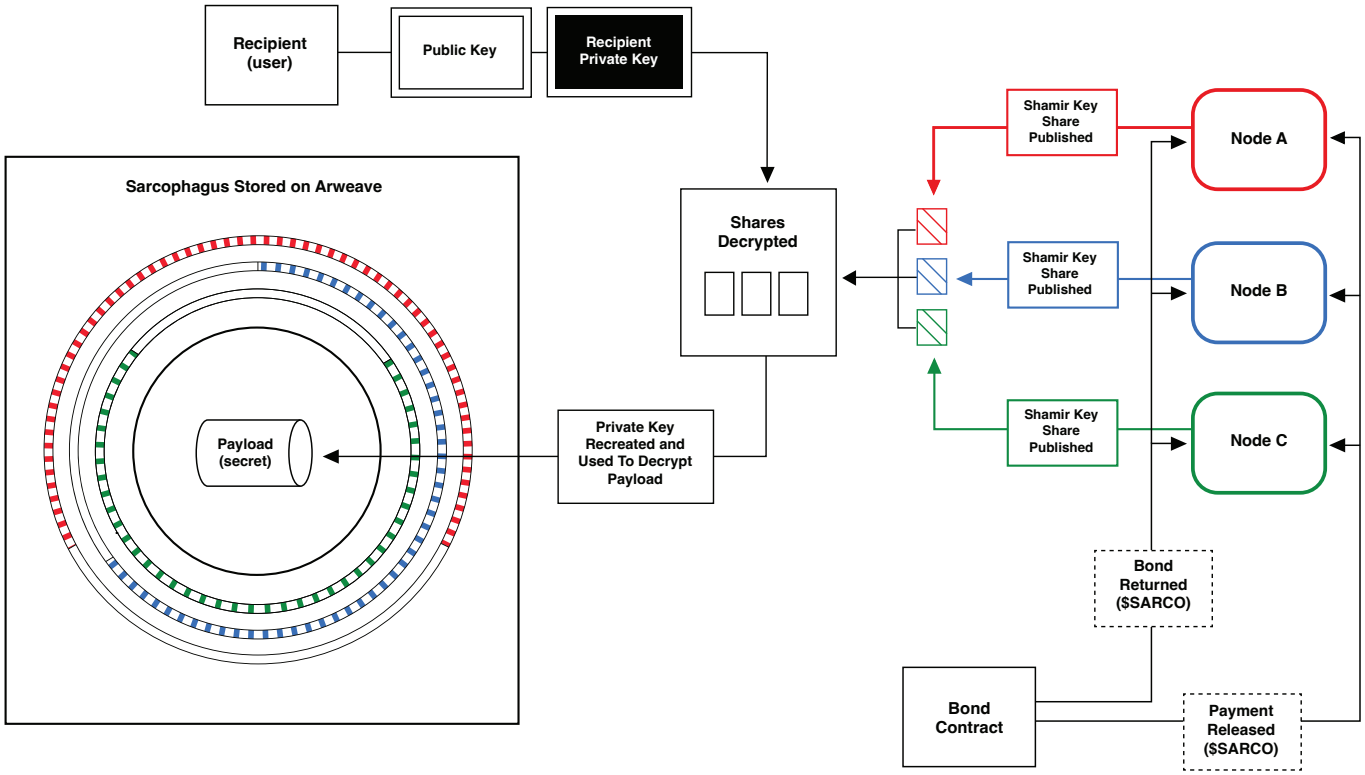
III. BURYING THE SARCOPHAGUS

PAYLOAD RELEASE / RESURRECTION

If the user fails to attest to the contract within the time period they set either during creation or during the most recent attestation, the nodes will publish their portion of the release key to the chain.

When the nodes publish this key, the recipient of the sarcophagus now has the ability to download the payload file from the application. The payload still remains encrypted where it is stored (on Arweave) but is now available to be decrypted by the designated recipient only. There are no fees or transactions required to download the payload.

After the payload release keys have been published by the nodes, their job is considered complete and they are paid the fees from the contract, and their bond is released back into the node wallet.



IV. 2OF3 SARCOPHAGUS RESURRECTION

SARCOPHAGUS DAO & GOVERNANCE

The Sarcophagus network and its parameters are managed via the Sarcophagus DAO (decentralized autonomous organization), which leverages the Aragon DAO management platform for all decision making and financial transitions.

The DAO owns all of the $SARCO tokens that are not in circulation, as well as the funds required to employ the Sarcophagus team. In order to participate in the governance operations of the DAO, $SARCO token holders must place their tokens in the governance contract available here: https://govern.sarcophagus.io/. When tokens are in this contract they are unable to be used for any other operation (paying fees to nodes, running a node, etc). While they are unable to be used for other operations, tokens in the governance contract can be withdrawn at any time.

Any desired changes on the protocol necessitates a proposal to be approved. The requirements for a passing proposal is a quorum of 10% participation of the governance token supply and a majority vote of 60% approval. All parameters that the DAO operates under can also be changed as a result of a successful vote.

Sarcophagus DAO leverages a modularized subDAO structure to enable teams to provide value and growth in targeted areas as well as to enable each subDAO with its own treasury to manage funds. This creates an agile framework for working teams to devote time to specific tasks and doesn't require full DAO proposal approvals. Each subDAO charter defines the working scope of the subDAO and all subDAOs can be disbanded at any time, subject to a vote of the main DAO.

SARCO TOKEN

The ERC-20 $SARCO token is the primary unit of exchange in the network. It is non-upgradable, has a fixed supply of 100,000,000 tokens, and has 18 decimal places (like Ethereum itself).

The token contract is available here:
https://etherscan.io/token/0x7697b462a7c4ff5f8b55bdbc2f4076c2af9cf51a

$SARCO is used in several ways within the network:

- By the user to pay for dead man's switch creation / monitoring / resurrection services.
- By the node to bond into contracts allowing accepting new customers and to receive payment for monitoring / resurrection services.
- By the DAO to vote for operations and to change network variables and to receive fees from the creation and attestation of sarcophagi.

GOVERNANCE VOTING INCENTIVE DISTRIBUTION

The Sarcophagus DAO takes a 1% fee from all sarcophagus creation and attestation actions. This fee is taken from the user, and is debited at the same time as their payment to the node(s) they employ. This 1% fee is a network variable and can only be changed with a vote from the DAO.

In order to fairly distribute network fees to active participants within the governance application, it is imperative that a regime be instituted to incentivize all participants, not just those with large token holdings.

Generally in token-weighted voting systems, only the largest players are incentivized to participate. This is an issue because even small players can have an excise impact on the ongoing operation of the network. Collections of smaller, but more engaged players are the lifeblood of new technology and it is important to allow them to be rewarded for their contributions.

In order to reward smaller participants in the governance contract, the Sarcophagus DAO has instituted a fee distribution model that does not punish lack of action, but rather rewards positive action for any participant, regardless of their $SARCO holdings.

This system works by calculating the number of votes possible within a given epoch, and then rewards each participant their pro-rata portion of the network fees during the period, multiplied by their participation rate. If all participants vote (aye or nay) for each vote within the period, the fee distribution will be exactly pro-rata. However if a participant only votes in half of the proposals, they will only receive half of the pro rata fees.

The fees left over from the participants that failed to voice their opinion in all of the votes are then distributed equally among all participants with a voting record of 100%. This scheme allows for a more egalitarian fee distribution model, incentivizing smaller participants to spend ETH on gas to vote, but also does not punish participants that are unable to vote for any reason.

CONTACT

https://sarcophagus.io

https://t.me/sarcophagusio

https://discord.gg/5XkvsEmMab

nospam@sarcophagus.io

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: OpenPGP.js v4.10.7
Comment: https://openpgpjs.org
xjMEX0RtehYJKwYBBAHaRw8BAQdAHT2CAePaeVHyMtIMbGsN8aaATyz5DkQ0TNVpV3SRSGjNLW5vc3BhbUBzYXJjb3BoYWd1c
y5pbyA8bm9zcGFtQHNhcmNvcGhhZ3VzLmlvPsKPBBAWCgAgBQJfRG16BgsJBwgDAgQVCAoCBBYCAQACGQECGwMCHgEAI
QkQHLZa1vaeT7gWIQRw0cZDu2CCF27JK3UctlrW9p5PuGyqAQDp9bpd1gd6/BU3s1vgAf9RTXp+0zk4gRLeIE3jQxAHRAEA
xJ5T2DbSNuwuyKV3NDxhfbjhtwRCpPYfoWrOXMrntwHOOARfRG16EgorBgEEAZdVAQUBAQdAVawc58xyjpS+j/7zunaSbGFx
Uj2y51VbzcONcy7wF0DAQgHwngEGBYIAAkFAl9EbXoCGwwAIQkQHLZa1vaeT7gWIQRw0cZDu2CCF27JK3UctlrW9p5PuBq3AQ
Dqds+8Kf2wxfrqZWw0VpGrxeuv5s7+qS0PhedIVP7z/wD9EFqanhQK7y7E9/pR1Mw4/KIWYeSXCayoOwSIpPn5ZgU==J1Ok
-----END PGP PUBLIC KEY BLOCK-----