

BIP-32: Full Version

Proposer

Halborn, Inc. and the Beanstalk Seraph Committee

Halborn is a team of 100+ award-winning ethical blockchain hackers who focus on securing their clients' full stack end-to-end. Trusted by Solana, Near, BAYC, Ava Labs and many more. Halborn has completed various security audits of Beanstalk and continues to audit BIPs as they are developed.

Seraph is a non-custodial blockchain security notary (BSN). A BSN is a cybersecurity professional who serves an organization (centralized or decentralized) as a third party witness to the signing of important on-chain actions. A BSN's main purpose is to deter fraud and prevent attacks.

Proposer Wallet: [\[UPDATE\]](#)

Summary

- Hire Halborn, Inc. for their blockchain security notary product called Seraph to deter new attacks on the protocol;
- Pay 10,000 USDC per month for Seraph for 6 months;
- Form the Beanstalk Seraph Committee (BSC), the group that has discretion over Runbooks for Seraph-protected functions and removing Seraph in cases where it must be removed for the security or censorship resistance of Beanstalk; and
- Make various updates to Beanstalk governance detailed below, including updating the Beanstalk Community Multisig (BCM) Process, the Beanstalk Immunefi Committee (BIC) Process, the Immunefi Bug Bounty Program and the Beanstalk DAO Disclosures.

Links

- [BIP-32 GitHub PR](#)
- [Safe Transaction](#) [\[UPDATE\]](#)
- GitHub Commit Hash: [\[UPDATE\]](#)

Problem

Security is paramount to Beanstalk's success. Beanstalk is a complex DeFi protocol that can be vulnerable to different attacks that are not always caused by code flaws or detected during audits. The loss associated with DeFi protocol exploits gets bigger day by day. The Beanstalk DAO should seek to avoid another hack as best as possible.

Under the current Beanstalk governance structure, only a single multisig (the BCM) needs to be compromised in order to corrupt Beanstalk. Implementing Seraph with no other upgrades to Beanstalk governance would make Beanstalk less resistant to censorship.

Beanstalk governance, the BCM Process, the BIC Process, the Immunefi Bug Bounty Program and the Beanstalk DAO Disclosures can all be updated to reflect the current state of Beanstalk and its ecosystem, and in particular the implementation of Seraph.

Proposed Solution

Seraph

A governance structure with a separate multisig that can remove Seraph can mitigate this concern and (combined with Seraph) increase the security and censorship-resistance of Beanstalk. We propose implementing Seraph into Beanstalk as an extra line of defense against hacks or other destructive actions to Beanstalk, and incorporate other complementary changes to Beanstalk governance.

We propose paying 10,000 USDC per month for Seraph for 6 months from the Beanstalk Farms budget.

Protected Functions

The Seraph platform can protect up to 25 of the highest risk smart contract functions embedded in Beanstalk. The Seraph code modifier protects 7 of the highest risk owner functions in Beanstalk:

- `diamondCut`
- `whitelistToken`
- `dewhitelistToken`
- `unpause`
- `transferOwnership`
- `createFundraiser`
- `addUnripeToken`

Seraph provides 24/7/365 services to review, analyze, and permit or reject any calls to these functions according to the appropriate Runbook(s).

Runbooks

Seraph notaries are required to process any function calls according to the specific notary Runbooks of rules and procedures.

Each of the 7 Seraph-protected functions has a unique Runbook which establishes the rules and procedures for Seraph notaries to process transactions that call the functions and the priority and risk of each function. The Runbooks for each protected function shall remain confidential between Halborn and the BSC.

The details about transactions that call protected functions and whether they have been reviewed, approved or rejected by the Seraph notary are publicly viewable via the [Seraph Dashboard](#).

Beanstalk Seraph Committee

We propose forming the BSC—an anonymous group of six reputable community members and Beanstalk core contributors. The BSC members were selected by Publius.

The Beanstalk Seraph Committee Multisig (BSCM) is the only wallet that can remove Seraph from Beanstalk. The BSC members are the only signers on the BSCM. The BSCM is a 4-of-6 multisig deployed using Safe.

The BSCM cannot call any of the owner functions of Beanstalk—only the BCM can. Publius attests that there is a minority of overlap between the signers on the BSCM and BCM.

Responsibilities

Seraph notaries have created the Runbooks in collaboration with the BSC to implement and activate the Seraph protections as effectively and safely as possible. The BSC has approved initial Runbooks for all protected functions.

The BSC can work with Halborn to update Runbooks at any time, but must sign a transaction verifying the new Runbooks for the change to be valid.

Seraph may be deactivated at any time via BIP. However, in instances where Halborn is unwilling to commit a passed BIP that removes Seraph, or where Seraph must be removed for the security or censorship resistance of Beanstalk, the BSC is responsible for doing so.

On the Seraph contract, the BSCM can call the `removeSeraph` function that initiates a 24 hour timelock. After the timelock elapses, the BSCM can call the `executeRemove1` function that removes Seraph from Beanstalk.

Best Practices

BSCM Signers shall follow the best practices outlined below. It is of paramount importance that Beanstalk limits key man risk by implementing best practices with respect to multisig key custody. Signers are expected to:

- Regularly check in with the rest of the BSCM and confirm access to their wallet;
- Maintain active communication regarding travel plans and availability in order to ensure that there are always enough Signers on call; and
- Acknowledge their Signer duties and processes.

In addition to the above expectations, Signers shall follow the BCM's wallet security best practices:

- Use a reputable hardware wallet like Trezor or Ledger;
- Use a fresh wallet that doesn't have any pre-existing transactions or balances on it;
- Set up a new passphrase on their hardware wallet device when selecting a new wallet to be the signing wallet; and
- Follow the standard self-custody best practice guide [here](#).

In the event that one or more Signers are compromised, unresponsive or attempting to violate the processes outlined in the **Responsibilities** section (or a Signer voluntarily chooses to be removed from the BSCM), the BSCM will rotate them out of the wallet and replace them with another Signer. Emergency changes to the m-of-n configuration of the BSCM that are made to protect Beanstalk do not require a BIP.

Anonymous Multisig Signers

Off-chain governance introduces significant risks related to security and censorship. The BSCM is designed to mitigate as many of those risks as possible by distributing the multisig keys across reputable community members and Beanstalk core contributors, and collectively implementing and adhering to BSCM best practices.

The most significant risk associated with off-chain governance is the potential corruption of the BSCM (and BCM). In order to minimize the chances of this, the Signers are anonymous. The anonymous Signers have been selected by Publius. Signers are anonymous to each other as well, apart from Publius.

Malicious Key Holder Risk

Under this structure, it's important to acknowledge the risk of anonymous key holders conspiring to attack Beanstalk. Because only Publius knows the identities of the anonymous Signers, Publius would be the main attack vector—if this malicious actor were to compromise Publius before conspiring to attack Beanstalk, they could be reasonably sure that their identity would never be revealed.

In order to mitigate this attack vector, the BSCM will institute the following process whenever the m-of-n configuration of the BSCM is changed:

- Publius will publish a hash of the list of Signers and their corresponding wallets on-chain; and
- Publius will share the list of Signers and their wallets with their personal legal counsel, to be released in the event that Publius is compromised such that they cannot publish the list themselves. This makes Publius and their personal legal counsel the only parties with access to the list.

The BSC Process can be read [here](#).

Process Amendments

Beanstalk Governance

We propose the following list of changes to Beanstalk governance:

- The Voting Period opens when the Snapshot proposal for a BIP can be voted on and ends after 7 days or once a two-thirds supermajority is reached;
- A Stalkholder's vote for a given BIP is counted as their Stalk plus Grown Stalk at the beginning of the Voting Period that still exists (Voting Stalk);
- If at the end of the Voting Period:
 - Less than or equal to half of total Voting Stalk votes For the BIP, it fails, or
 - More than half of total Voting Stalk votes For the BIP, it passes;
- If at any time 24 hours or more after the beginning and before the end of the Voting Period more than two-thirds of the total Voting Stalk votes For the BIP, the BCM can execute the BIP on-chain;
- Voting Stalk is also used to count votes in all other Beanstalk governance proposals where Stalkholders vote (i.e., BOPs, BFCPs and BSPs);
- The proposer must have 0.1% of total Voting Stalk at the end of a Voting Period where Stalkholders vote (i.e., BOPs, BFCPs and BSPs);
- Only BIPs can (1) change the Beanstalk protocol, (2) mint Beans and/or (3) change Beanstalk governance;
- BOPs are proposals for the DAO to agree on processes that do not involve changes that can only be made in BIPs.
- Voting choices for a BOP are For and Against;
- BOPs and BFCP-Bs must have >35% of total Voting Stalk vote For and a majority of participating Voting Stalk vote For at the end of the Voting Period in order to pass; and
- BFCP-As must have >25% of total Voting Stalk vote For and a majority of participating Voting Stalk vote For at the end of the Voting Period in order to pass.

The updated Beanstalk governance process can be read in the new Proposals documentation [here](#).

BCM Process

We propose the following list of changes to the BCM Process:

- Integrate all of the changes included in the above Beanstalk Governance section;
- Add the `addUnripeToken` and `addFertilizerOwner` functions to the list of owner functions for completion;
- Require a BIP for non-emergency changes to the m-of-n configuration of the BCM, but allow emergency changes to the m-of-n configuration without a BIP in order to remove rogue Signers before a replacement Signer can be selected;
- Create significantly more clear and permissionless BIP and BOP proposal processes;
- Require that written proposals for BIPs have a **Proposer** section that at minimum includes a verified message signature uploaded to Arweave;
- Document the process for uploading verified message signatures to Arweave for both BIP proposers and BCM Signers;
- Require that written proposals for BIPs that call `diamondCut` have a **Contract Changes** section that at minimum lists the facets and Init contract addresses that `diamondCut` calls; and
- Require that written proposals for BIPs that call `diamondCut` have a **Beans Minted** section that described the number of Beans minted by the execution of the `diamondCut` ;
- Make it clear that BIPs may have multiple transactions if made explicit in the written proposal;
- Add relevant BCM processes for verifying and executing Beanstalk Immunefi Responses (BIRs);
- Remove the requirement for Signers to rotate wallets every 2 months;
- More thoroughly document the processes for BCM Signers to verify and sign transactions;
- Specify that in cases where functions must be removed immediately to protect Beanstalk, BCM Signers are not required to submit a verified message signature on Arweave; and
- Specify that in no instance shall the majority of the BCM keys be held by Beanstalk Farms contributors.

The updated BCM Process can be read [here](#). The guides for uploading verified message signatures to Arweave can be read [here](#).

BIC Process

The following is a list of proposed changes to the BIC Process:

- Add Fix section that describes the process that is followed in order to fix a reported issue;
- Specify that the potential economic damage determination is only required in BIRs for bug reports qualified as High or Critical Impact;
- Clarify the process that the BIC follows after a fix is implemented and a determination has been made by the BIC;
- Require that the Safe transaction link and **Beans Minted** sections appear in the BIR on Snapshot;
- Make it explicit that the BIC has the ability to update the Immunefi Bug Bounty Program in the following ways, without the need for a BOP or BIP:
 - Adding new assets as in-scope, including contracts that have been audited by Halborn and/or have the potential to attract a meaningful amount of Beans/BDV;
 - Updating the list of pull requests for BIPs whose code has been audited by Halborn but is not yet deployed;
 - Adding documentation and links; and
 - Updating language to improve clarity;
- Form the BIC Multisig (BICM), a 4-of-6 Safe multisig on which BIC members serve as signers;
- Specify that the BICM custodies ownership of the contract that updates the number of Beans remaining approved by the DAO to be minted for bug bounties;
- Create a list of BICM backup signers:
 - Al Bean;
 - Mr Mochi; and
 - guy;
- Allow the BICM to rotate signers to a backup signer without the need for a BOP or BIP.

The updated BIC Process can be read [here](#).

Immunefi Bug Bounty Program

The following is a list of proposed changes to the Immunefi Bug Bounty Program:

- Specify that Beans/BDV at risk are the primary consideration when determining practicable economic damage;
- Specify that Circulating non-Bean assets at risk may qualify for a maximum of up to 50% of the potential reward outlined in the Rewards by Threat Level section, as determined by the BIC; and
- Specify that unexpected outcomes (like loss of funds) due to misuse of Pipeline do not qualify as valid bug reports.

The updated Immunefi Bug Bounty Program can be read [here](#).

Disclosures

The following is a list of proposed changes to the Beanstalk DAO Disclosures:

- Add the following disclosure about Seraph:

MOST OWNER FUNCTIONS OF THE BEANSTALK CONTRACT ARE PROTECTED BY SERAPH, A BLOCKCHAIN SECURITY NOTARY SERVICE, IMPLEMENTED BY HALBORN, INC. THERE IS NO GUARANTEE THE RUNBOOKS FOR SERAPH PROTECTED FUNCTIONS ARE FOLLOWED OR THAT THE RUNBOOKS HELP PROTECT BEANSTALK. THERE IS NO GUARANTEE THAT SERAPH PROTECTIONS ARE ONLY REMOVED WHEN APPROPRIATE.

The Beanstalk DAO implemented Seraph into Beanstalk, a blockchain security notary service offered by Halborn, Inc. Every function protected by Seraph requires a Runbook. Runbooks are the set of rules and procedures for Seraph to process transactions that call protected functions.

Seraph notaries created and maintain the Runbooks in collaboration with the Beanstalk Seraph Committee (BSC) to activate and implement the Seraph protections as effectively and safely as possible. The BSC's other responsibility is serving as signers on the multisig that is the only wallet that can remove Seraph protection from Beanstalk. The BSC members are anonymous and were selected by Publius. This process was approved via governance.

Seraph introduces significant risks related to security and censorship. There is no guarantee that:

- The Runbooks approved by the BSC are followed by Seraph notaries;
 - Any of the Runbooks that are followed protect Beanstalk;
 - The BSC will remove Seraph protections when necessary for the security or censorship resistance of Beanstalk; or that
 - The BSC does not remove Seraph protections when it is not beneficial to Beanstalk.
- Add the following disclosure about the Beanstalk Subgraph:

THE APP.BEAN.MONEY FRONTEND DEPENDS ON THE BEANSTALK SUBGRAPH FOR DISPLAYING VARIOUS ON-CHAIN DATA. THERE IS NO GUARANTEE THAT SUBGRAPH DATA IS ACCURATE.

The Beanstalk UI hosted at app.bean.money depends on the Beanstalk Subgraph for displaying data, particularly on the Market and Analytics pages. The Beanstalk Subgraph is primarily developed and deployed by Beanstalk Farms.

By default the Beanstalk UI uses a version of the subgraph hosted by Beanstalk Farms, which can be censored. The subgraph that the Beanstalk UI uses can be adjusted in the settings.
 - Add a note about the Beanstalk SDK being unaudited in Disclosure #19; and
 - Allow Beanstalk Farms to modify the Disclosures that pertain to Beanstalk development tooling as they change without the need for a BIP or BOP.

The updated Beanstalk DAO Disclosures can be read [here](#).

Rationale

Seraph

Technical analysis of attacks on DeFi protocols over the last 12 months indicate that both the volume and loss-value of attacks on protocols are increasing substantially.

With Seraph, Beanstalk can deter additional attacks on the protocol and disincentivize attackers by making it more difficult and costly to conduct an attack. The fee for Seraph is substantially lower than the projected operational and reputational costs associated with an additional attack. By activating Seraph, Beanstalk assets and contracts are protected such that complex attacks may be prevented before they can be executed on-chain.

As Beanstalk returns to on-chain governance, the Beanstalk DAO and BSC can continue to work with Seraph as an extra layer of defense.

Beanstalk Seraph Committee

Implementing Seraph with no other upgrades to Beanstalk governance would make Beanstalk less resistant to censorship. By implementing Seraph and alongside the formation of the BSC, both the BCM and the BSCM must be compromised in order to corrupt Beanstalk, without compromising censorship resistance before on-chain governance is able to be reimplemented.

Runbooks for each protected function remain confidential between Halborn and the BSC in order to mitigate potential manipulation of Beanstalk by malicious actors.

BSCM Signers are anonymous to minimize the potential corruption of the BSCM from an outside party.

Process Amendments

The other proposed changes to Beanstalk governance (such as the introduction of Voting Stalk, the requirement to meet the proposer Stalk threshold at the end of the Voting Period, etc.) all complement or supplement the introduction of Seraph.

The proposed changes to the BCM Process significantly increase the quality of documentation for and the permissionlessness of the BIP and BOP proposal processes. The BCM Process must be updated to reflect the Beanstalk governance changes proposed in this BIP.

The proposed changes to the Immunefi Bug Bounty Program further refine what bug reports are considered valid and how the BIC determines practicable economic damage, which improves the bug reporting experience for Immunefi whitehats. The proposed changes to the BIC Process significantly increase the quality of documentation for the BIC operating processes and gives the BIC more flexibility in updating the Immunefi Bug Bounty Program. The formation of the BICM allows the number of remaining Beans approved by the DAO to be minted for bug bounties to be queryable on-chain.

The proposed changes to the Beanstalk DAO Disclosures reflect the implementation of Seraph and the current state of Beanstalk development tooling.

Contract Changes

The Δ symbol indicates that there is a proposed change in functionality.

Diamond Cut Facet

The following `DiamondCutFacet` is being removed from Beanstalk:

- `0xdfeff7592915bea8d040499e961e332bd453c249`

The following `DiamondCutFacet` is being added to Beanstalk:

- UPDATE

`DiamondCut` Function Changes

Name	Selector	Action	Type	Δ
<code>diamondCut</code>	<code>0x1f931c1c</code>	Replace	Call	✓

`DiamondCut` Event Changes

None.

Whitelist Facet

The following `WhitelistFacet` is being removed from Beanstalk:

- `0xaea0e6e011106968adc7943579c829e49efddad0`

The following `WhitelistFacet` is being added to Beanstalk:

- UPDATE

`WhitelistFacet` Function Changes

Name	Selector	Action	Type	Δ

dewhitelistToken Name	0x86b40a1b Selector	Replace Action	Call Type	✓ Δ
whitelistToken	0xd8a6aafe	Replace	Call	✓

WhitelistFacet Event Changes

None.

Pause Facet

The following PauseFacet is being removed from Beanstalk:

- 0xeab4398f62194948cb25f45fee4c46fae2e91229

The following PauseFacet is being added to Beanstalk:

- UPDATE

PauseFacet Function Changes

Name	Selector	Action	Type	Δ
upause	0x3f4ba83a	Replace	Call	✓
pause	0x8456cb59	Replace	Call	

PauseFacet Event Changes

None.

Ownership Facet

The following OwnershipFacet is being removed from Beanstalk:

- 0x5d45283ff53aabdb93693095039b489af8b18cf7

The following OwnershipFacet is being added to Beanstalk:

- UPDATE

OwnershipFacet Function Changes

Name	Selector	Action	Type	Δ
transferOwnership	0xf2fde38b	Replace	Call	✓
claimOwnership	0x4e71e0c8	Replace	Call	
owner	0x8da5cb5b	Replace	View	
ownerCandidate	0x5f504a82	Replace	View	

OwnershipFacet Event Changes

None.

Fundraiser Facet

The following FundraiserFacet is being removed from Beanstalk:

- [0x538c76976ef45b8ca5c12662a86034434bfc7a8e](#)

The following FundraiserFacet is being added to Beanstalk:

- UPDATE

FundraiserFacet Function Changes

Name	Selector	Action	Type	Δ
createFundraiser	0x4b4e8d9a	Replace	Call	✓
fund	0x43c5198e	Replace	Call	
fundingToken	0xc869c1eb	Replace	View	
fundraiser	0xce133450	Replace	View	
numberOfFundraisers	0x6299a9af	Replace	View	
remainingFunding	0x0d1a844c	Replace	View	
totalFunding	0x6ee66ddf	Replace	View	

FundraiserFacet Event Changes

None.

Fertilizer Facet

The following FertilizerFacet is being removed from Beanstalk:

- [0xfc7ed192a24fab3093c8747c3ddbe6cacd335b6c](#)

The following FertilizerFacet is being added to Beanstalk:

- UPDATE

FertilizerFacet Function Changes

Name	Selector	Action	Type	Δ
addFertilizerOwner	0x8cd31ca0	Replace	Call	✓
claimFertilized	0x83e08888	Replace	Call	
mintFertilizer	0x0bfca7e3	Replace	Call	

payFertilizer Name	0xd47aee59 Selector	Replace Action	Call Type	Δ
balanceOfBatchFertilizer	0x304ec65d	Replace	View	
balanceOfFertilized	0xb6f42085	Replace	View	
balanceOfFertilizer	0x1799b3b2	Replace	View	
balanceOfUnfertilized	0x1edb6be1	Replace	View	
beansPerFertilizer	0x9bb4e35a	Replace	View	
getActiveFertilizer	0xdc6ba285	Replace	View	
getCurrentHumidity	0x39448802	Replace	View	
getEndBpf	0xc85951a1	Replace	View	
getFertilizer	0x9c45a1d5	Replace	View	
getFertilizers	0x34af5416	Replace	View	
getFirst	0x1e223143	Replace	View	
getHumidity	0x29130a66	Replace	View	
getLast	0x4d622831	Replace	View	
getNext	0xf4a057e2	Replace	View	
isFertilizing	0x6ae1c014	Replace	View	
remainingRecapitalization	0x4a16607c	Replace	View	
totalFertilizedBeans	0x4f9a9678	Replace	View	
totalFertilizerBeans	0xf9c4ebde	Replace	View	
totalUnfertilizedBeans	0xa3ef48c9	Replace	View	

FundraiserFacet

Event Changes

None.

Beans Minted

None.

Effective

Effective immediately upon commit.