STAYSAFU AUDIT

SECURITY ASSESMENT: FEBRUARY 27TH, 2022

AVALANT

TABLE OF CONTENTS

SUMMARY

*This report has been prepared for **Avalant** to discover issues and vulnerabilities in the source code of the **Avalant** project as well as any contract dependencies that were not part of an officially recognized library.*

The audit is based on the code of the following gist link (Antgold.sol, Avalant.sol and Suga.sol) :

https://gist.github.com/avalantnft/6aae47950dc149839b368da7c6858036

*A comprehensive examination has been performed, utilizing Static Analysis, Manual Review, and **Avalant** Deployment techniques. The auditing process pays special attention to the following considerations:*

- Testing the smart-contracts against both common and uncommon attack vectors
- Assessing the codebase to ensure compliance with current best practices and industry standards
- Ensuring contract logic meets the specifications and intentions of the client
- Cross referencing contract structure and implementation against similar smart-contracts produced by industry leaders
- Through line-by-line manual review of the entire codebase by industry experts

OVERVIEW

VULNERABILITY SUMMARY

UNDERSTANDING

The **Avalant** project is a **P2E** (Play To Earn) videogame based on NFTs. Players own **Avalant** (or ant) NFTs which work for them and produce **antgold** (**ANTG**). The deeper ants are in the ground, the more **antgold** players earn. Players can then trade or stake this **antgold** for **suga** (**SUGA**). The **suga** can then be used to gain energy to dig deeper or to fight bosses that give access to new depths.

Avalant.sol :

Avalant is the NFT that powers the whole game. It features a whitelist mechanism, a maximum total supply (**10 000** ants), minting mechanisms for both whitelisted and not whitelisted players including fees (**0.8 AVAX** for whitelisted players, **1.2 AVAX** for others when buying an ant), floor-changer and name-changer (taxed by **500 antgold**) for ants and a royaltee fee distribution (**4%** when buying a new ant). The NFT also features a system of boxes for the presale and the public sale via an attached smart-contract (**AVALANT_BOXES_CONTRACT**).

Antgold.sol :

Antgold is an **ERC20** token used to reward players. It implements staking and claiming ants, **suga**/**antgold** exchange and manual airdrop from admins. The income from ant staking is calculated according to the following method:

If the ant has been staked for less than 10 days, the staking income is equal to the basic income per ant per day per floor multiplied by the number of floors plus the basic income of a staked ant.
(BASE_ANTG_BY_ANT_PER_DAY_PER_STAGE * colonyStage +BASE_ANTG_BY_ANT_PER_DAY)

If the ant has been staked for more than 10 days, the staking income is the same but **5%** is added.

Suga.sol :

Suga(**SUGA**) is an **ERC20** token used in-game by players. It implements staking and claiming **antgold**, **antgold**/**suga** exchanges and airdrop from admins.

Antgold/**suga** exchanges are influenced by the **ANTG_RATIO**, which is updateable by the admin.

The income from **antgold** staking is calculated according to the following method :

The staking income is equal to the **antgold**-per-day ratio multiplied by the number of days **antgold** has been staked.

(_antgStaked * ANTG_STAKE_DAY_RATIO) *((block.timestamp - antgStakedFrom[account]) * 10000000000) / 86400) /10000000000;)

The admin can airdrop **suga** within the **MAXSUPPLY** (25_000_000_000) limit.

Avalant

The following elements are extracted from the analysis of the codebase and the tests carried out on the **Avalant.sol** smart-contract.

OWNERSHIP

Here is a non-exhaustive list of what the smart-contract administrator (represented by the **DEFAULT_ADMIN_ROLE**) can do.

Feature	Able to modify / to do	Details
Modifying ant minting price	**Yes**	
Modifying presale restricted number	**Yes**	
Opening the ant minting for	**Yes**	

presale		
Modifying royalty fees	**Yes**	
Adding late players to whitelist	**Yes**	
Opening the ant minting for public	**Yes**	
Modifying ant contracts addresses	**Yes**	Admin is able to change the antgold, suga and bosses contract addresses

FINDINGS

Unclear error messages

Severity: N/A

Some error messages are unclear. For example, line 107 '"**Ant escaped the box, sorry**"' does not explicitly specify why the **openBox** function failed. We recommend using the clearest possible error messages for maintenance/development and data processing. The following table contains corrections for unclear error messages.

Before	After
"Ant escaped the box, sorry"	**“Can’t open box, maximum supply ”**
“Too many ants"	**“Maximum ant/player reached”**
"Ant is too tired to dig"	**“Ant’s rest time is not reached”**
"Blocked by boss"	**“Can’t dig : the boss is**

	still alive”

Centralization of major privileges

Severity: Medium

The administrator of the smart-contract has major privileges over it (he can change royalties fee, set a new royaltee address, change ant contracts addresses, open or close the ant minting). This can be a problem, and we recommend at least to use a multi-sig wallet for the admin account, and at best to establish a community governance protocol to avoid such centralization. Overall, this problem is common and not exclusive to **Avalant.sol** smart-contract.

Antgold

The following elements are extracted from the analysis of the codebase and the tests carried out on the **Antgold.sol** smart-contract.

OWNERSHIP

Here is a non-exhaustive list of what the smart-contract administrator (represented by the **DEFAULT_ADMIN_ROLE**) can do.

Feature	Able to modify / to do	Details
Minting/ Airdroping antgold	**Yes**	
Modifying ant contracts addresses	**Partially**	Admin is able to change the suga smart-contract address

Separation of the claim mechanism into two functions

Gas optimization

In order to optimize the gas costs of the claim, the calling of several functions should be avoided as much as possible. It is possible to delete the **claimAntGold** function and simply set the **_claimAntGold** function to external.

No threshold for minting and airdropping functions

Severity: Minor

The smart-contract admin can mint a potentially infinite amount of **antgold** via the **mint** function. By extension, the **airdrop** function that uses the **mint** function is not subject to a limit either. We strongly recommend adding a threshold and a cooldown to the **airdrop** function.

Centralization of major privileges

Severity: Medium

The administrator of the smart-contract has major privileges over it (he can airdrop/mint **antgold** without any threshold or cooldown and change the **suga** contract address). This can be a problem, and we recommend at least to use a multi-sig wallet for the admin account, and at best to establish a community governance protocol to avoid such centralization. Overall, this problem is common and not exclusive to **Antgold.sol** smart-contract.

Suga

The following elements are extracted from the analysis of the codebase and the tests carried out on the **Suga.sol** smart-contract.

OWNERSHIP

Here is a non-exhaustive list of what the smart-contract administrator (represented by the **DEFAULT_ADMIN_ROLE**) can do.

Feature	Able to modify / to do	Details
Update the antgold/suga ratio	**Yes**	
Airdropping suga	**Partially**	Admin can airdrop suga within the **MAX_SUPPLY** limit
Claim suga	**Yes**	

for people		

Centralization of major privileges

Severity: Medium

The administrator of the smart-contract has major privileges over it (he can modify the **antgold**/**suga** ratio and claim **suga** for everyone). This can be a problem, and we recommend at least to use a multi-sig wallet for the admin account, and at best to establish a community governance protocol to avoid such centralization. Overall, this problem is common and **Suga.sol** presents a satisfactory level of centralization.

CONCLUSION

The **Avalant** project has a very good safety record. No major security issue has been found, the problems found are low or medium severity issues. The only points that should be improved is the centralization of privileges, and the contract code's abidance to best practices.

DISCLAIMER

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to the Company in connection with the Agreement.

This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement.

This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes without **StaySAFU**'s prior written consent. This report is not, nor should be considered, an “endorsement” or “disapproval” of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any “product” or “asset” created by any team or project that

contracts **StaySAFU** to perform a security assessment.This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance. This report should not be used in any way to make decisions around investment or involvement with any particular project.

This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a
high level of ongoing risk.

StaySAFU's position is that each company and individual are responsible for their own due diligence and continuous security. **StaySAFU**'s goal is to help reduce the attack vectors and the

high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.