Smart Contract Audit

Alaska Gold Rush

Table of Contents

Summary

Project summary

Language	Solidity
Codebase	undisclosed
Commits	55aac70aef7ec38f213065fc5add2477f4629b46

Scope

ID	File
AGR	contracts/AlaskaGoldRush.sol
AB	@delegatecall/utils/contracts/AntiBot.sol
WT	@delegatecall/utils/contracts/Withdrawable.sol
VT	@delegatecall/vesting/contracts/IVestable.sol
WST	@delegatecall/utils/contracts/ERC20/WithSupervisedTransfers.sol

Findings

AGR-01 | Excessive access control [Resolved]

Impact: Low

Description

The smart contract uses both owner and role-based access control schemes. This is excessive as admin role is equivalent to the owner.

Recommendation

Use role-based control everywhere and remove the reference to Ownable.sol

Comments

The team resolved this issue.

AGR-02 | Unnecessary transfer throttling [Resolved]

Impact: Low

Description

`transactionThrottler` is introduced as a way to mitigate sandwich attacks. To achieve that goal, it's enough to apply it to the `transferFrom` function. Throttling transfers is unnecessary.

Recommendation

Don't apply the `transactionThrottler` modifier to the transfer function.

Comments

The team resolved this issue.

AB-01 | Denial of service [Resolved]

Impact: Medium

Description

ERC-20 tokens allow anyone to call `transferFrom` function with amount 0 even if no allowance was previously given. An attacker can observe the mempool and frontrun all legitimate `transferFrom` calls with their own `transferFrom` call, effectively causing all transactions to be reverted.
This is considered medium impact because antibot functionality can be disabled.

Recommendation

Ignore all `transferFrom` function calls where amount equals zero.

Comments

The team resolved this issue.

WT-01 | Missing zero address validation [Resolved]

Impact: Low

Description

The withdrawETH function can be mistakenly called with no address specified, which would default to the zero address, sending all ether on the smart contract to the zero address and making it irrecoverable.

Recommendation

Check that the address is not zero.

Comments

The team resolved this issue.

Disclaimer

The smart contracts given for audit have been analyzed by the best industry practices at the date of this report, with cybersecurity vulnerabilities and issues in smart contract source code, the details of which are disclosed in this report (Source Code); the Source Code compilation, deployment, and functionality (performing the intended functions).

The report contains no statements or warranties on the identification of all vulnerabilities and security of the code. The report covers the code submitted to and reviewed, so it may not be relevant after any modifications. Do not consider this report as a final and sufficient assessment regarding the utility and safety of the code, bug-free status, or any other contract statements.

While we have done our best in conducting the analysis and producing this report, it is important to note that you should not rely on this report only — we recommend proceeding with several independent audits and a public bug bounty program to ensure the security of smart contracts. English is the original language of the report. The Consultant is not responsible for the correctness of the translated versions.

Smart contracts are deployed and executed on a blockchain platform. The platform, its programming language, and other software related to the smart contract can have vulnerabilities that can lead to hacks. Thus, Consultant cannot guarantee the explicit security of the audited smart contracts.