{%hackmd XdIXzOf5Ty2M3Uj1taFHIg %}

Alpha-Quark

Security Assessment

September 21st, 2020

For :
Kyle Kim @ Alpha Quark
contact@alphaquark.io

By :
Alex Papageorgiou @ CertiK
alex.papageorgiou@certik.org

PK: 970b96f66c3e3c77db81c874b3d02048a9cea893ded728f8365264d129490f0b

{%hackmd vb2ypisZSneY5y8y5ou-nw %}

Summaries

Project Summary

Project Name	Alpha Quark
Description	ERC-20 Token with Freezing and Locking mechanisms
Platform	Ethereum, Solidity
Codebase	GitHub Repository

Audit Summary

Delivery Date	Sep. 21, 2020
Method of Audit	Static Analysis, Manual Review
Consultants Engaged	1
Timeline	Sep. 1rst, 2020 - Sep. 21rst 2020

Vulnerability Summary

Total Issues	6
Total Critical	0
Total Major	0
Total Minor	0
Total Informational	6

Findings

ID	Title	Type	Severity
AQT-01	Inexistent `require` Messages	Coding Style	Informational
AQT-02	Getter Optimization	Optimization	Informational
AQT-03	Incorrect Emittance of `PauserRemoved` Message	Volatile Code	Informational
AQT-04	Redundant `delete` Operation	Optimization	Informational
AQT-05	Direct `length` Alteration	Coding Style	Informational
AQT-06	Combination of Setters	Optimization	Informational

AQT-01: Inexistent `require` Messages

Type	Severity	Location
Coding Style	Informational	AQToken.sol: L20, L30, L41, L52,L62, L76, L77, L86, L87, L97, L114, L118, L119, L133, L211, L219,L313, L346, L364, L378, L393, L407, L521, L550, L557, L641

Description:

Error messages provide insightful information on why a procedure failed, hence the importance of their existence.

Recommendation:

We advise that the linked `require` statements have an error message added to them.

Alleviations:

The team opted to consider our references and added error messages to all require statements.

AQT-02: Getter Optimization

Type	Severity	Location
Optimization	Informational	AQToken.sol: L123 - L130

Description:

The linked getter can directly return the clause of the `if` statement it checks.

Recommendation:

We advise the result of the comparison is directly returned.

Alleviations:

The team opted to consider our references and directly return the boolean value of the conditional `account == owner`.

AQT-03: Incorrect Emittance of `PauserRemoved` Message

Type	Severity	Location
Volatile Code	Informational	AQToken.sol: L174 - L176

Description:

The `PauserRemoved` message can be emitted for any user as the `renouncePauser()` function does not contain any guard that ensures the caller is already a pauser.

Recommendation:

We advise that proper access control is introduced.

Alleviations:

The team opted to consider our references and changed the access control for the linked function. Although the team used the `onlyOwner` modifier, the `onlyPauser` modifier should instead be used here.

AQT-04: Redundant `delete` Operation

Type	Severity	Location
Optimization	Informational	AQToken.sol: L604

Description:

As the data is replaced in the subceding line, the `delete` operation is redundant.

Recommendation:

We advise that he `delete` operation is removed.

Alleviations:

The team opted to consider our references and removed redundant code.

AQT-05: Direct `length` Alteration

Type	Severity	Location
Coding Style	Informational	AQToken.sol: L606

Description:

It is ill-advised to directly affect the `length` member of an array.

Recommendation:

We advise that a `pop()` operation is used instead.

Alleviations:

The team opted to consider our references and used the `pop()` operation.

AQT-06: Combination of Setters

Type	Severity	Location
Optimization	Informational	AQToken.sol: L549 - L561

Description:

The linked setters could be combined to a single one.

Recommendation:

We advise that a single setter function is implemented that accepts a boolean value as an argument to indicate "freeze" or "unfreeze" .

Alleviations:

The team opted to consider our references and combined the two setter functions into a single one, as described, as well as followed the same procedure with the respective events attached to the linked functions.