Security Audit of Vaulty a9943a by web3go.tech

Conclusion

Audit was done by the “Web3Go” team https://web3go.tech/ (https://web3go.tech/)
by Vladimir Smelov vladimirfol@gmail.com (mailto:vladimirfol@gmail.com)
https://www.linkedin.com/in/vladimir-smelov-25021669/ (https://www.linkedin.com/in/vladimir-smelov-25021669/)

In the final contract were not found:

- Backdoors for investor funds withdrawal by anyone.
- Bugs allowing to steal money from the contract.
- Other security problems.

Obvious errors or backdoors were not found in the contract.
The client was acknowledged about all secutiry notes below.

Scope

One specific update:

https://github.com/VaultyFinance/contracts/commit/a9943a9732b09c41b4ec96326f1b920e23c4ac8a (https://github.com/VaultyFinance/contracts/commit/a9943a9732b09c41b4ec96326f1b920e23c4ac8a)

The previous source code was already audited.

Methodology

1. Blind audit. Try to understand the structure of the code.
2. Find info in internet.
3. Ask quiestions to developers.
4. Draw the scheme of cross-contracts interactions.
5. Write user-stories, usage cases.
6. Run static analyzers

Find problems with:

- backdoors
- bugs
- math
- potential leaking of funds
- potential locking of the contract
- validate arguments and events
- others

Result

Critical

Not found.

Major

Warning

1. Potentialy dangerous decrement.

At:

- https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L174 (https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L174)
  you do

```
periodIndex--;
```

so if someone call claimForSelf twice then at the second call, the while condition will be skipped at:

- https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L152 (https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L152)
  and decreased periodIndex will be returned.

And at:
https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L222 (https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L222)
you set periodIndex back to the past.

There is a check at:

- https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L216 (https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L216)
  that should revert such transactions. But the check is not direct.

Recommendation.

Add something like

```
require(totalPeriods > periodIndex);
```

or get rid of strange

```
periodIndex--;
```

use `for` -loop for example over periodIndex.

Status.

NEW

2. While condition potentially can go out of block gas limit.

At:

- https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L152 (https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L152)

if someone will call the method for the first time or after a long time the while condition potentialy may process huge number of periods and go out of gas block limit.

Recommendation.

Create specific method to process specific number of periods to fit into gas block limit.

Status.

NEW

3. Lack of tests.

There is no test in the repo.
It makes the code easy to make a mistake and difficult to understand what is the expected flow of execution.

Recommendation.

Add tests.

Status.

NEW

Comment

1. Useless conversion uint64 to/from uint256.

At:

- https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L148 (https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L148)
- https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L222 (https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L222)

there is a conversion of the periodIndex from uint64 to uint256 and then back to uint64.

Recommendation.

Remove useless conversion.

Status.

NEW

2. Unclear setting daysClaimed to zero.

At:

- https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L170 (https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L170)
  you set claim.daysClaimed = 0
  but then at:
- https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L220 (https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L220)
  you set it back to non-zero where daysClaimed is taken from:
- https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L167 (https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L167)

it’s not clear why don’t just set at:

- https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L170 (https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L170)

```
claim.daysClaimed = daysClaimed
```

Recommendation.

Just set at:

- https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L170 (https://github.com/VaultyFinance/contracts/blob/a9943a9732b09c41b4ec96326f1b920e23c4ac8a/TokenVesting.sol#L170)

```
claim.daysClaimed = daysClaimed
```

Status.

NEW