Отчёт об аудите информационной безопасности смарт-контракта токена IFT (Ethereum)

D-D-S
Москва
2018

Содержание:

1. Общие условия

Отчёт составлен в ознакомительных целях и предоставляется «как есть» - без юридических гарантий безопасности.

2. Полученные сведения

Контракт предназначен для ведения реестра адресов акционеров, эмиссии токенов на их адреса, голосования акционеров и выплаты средств акционерам согласно распределению токенов среди них.

В каталоге представлено четыре файла:

- **`arbitrage.sol`** - основной контракт, реализующий функциональность токена, содержит описание ключевых переменных и функций
- **`Erc20.sol`** – контракт, обеспечивающий реализацию токен-стандарта ERC-20
- **`Ownable.sol`** – контракт, реализующий владение контрактом и передачу прав владения
- **`SafeMath.sol`** – вспомогательный контракт, использующийся для реализации безопасной арифметики в Ethereum

3. **Общие рекомендации**

`arbitrage.sol`

Основной контракт написан чисто и читаемо, однако в нём содержится код в комментариях - возможно, он использовался при тестировании. Рекомендуем избавиться от такого кода для улучшения читаемости и исключения ошибок при публикации (деплое) на блокчейн Ethereum.

Также было бы уместно использование более свежей версии компилятора (указана 0.4.24, рекомендуемая 0.5.1) и использование SafeMath во всех местах, где встречается арифметика uint256-переменных.

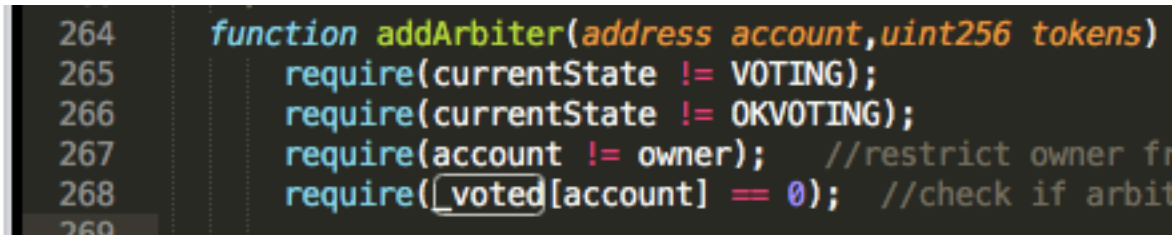
Временные интервалы для удобства восприятия рекомендуем переписать в виде **`k*days`**, где k - количество дней. Эта мера также позволит использовать переменные меньшего размера для хранения числа суток.

4.1 Обзор уязвимостей

`arbitrage.sol`
1:

Код не удовлетворяет требованиям заказчика в том смысле, что позволяет **добавить сам контракт в список акционеров**, а затем начислить на него токены. Контракт, в свою очередь, принадлежит владельцу. Таким образом, владелец, пусть и косвенно - через адрес контракта, но включил себя в число акционеров.

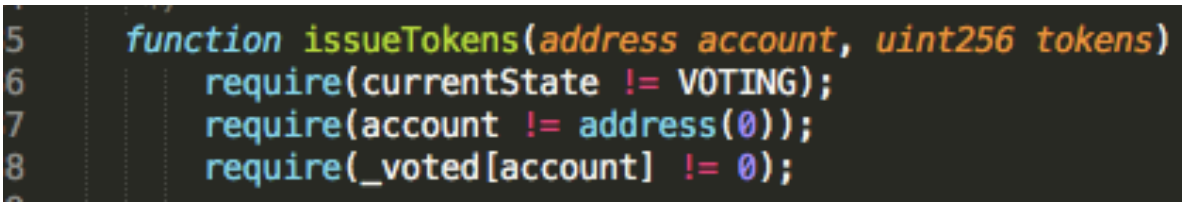
Функция `addArbiter` проверяет, сопадает ли адрес акционера с адресом владельца. Все другие адреса, в том числе адрес самого контракта, для неё допустимы (проверка `account != owner`).



Другие проверки в функции добавления будут пройдены наравне с остальными акционерами - как в силу логики кода, так и в силу особенностей языка Solidity.

Прочие условия функции `addArbiter` не ограничивают применимость метода.

В этом же отношении не имеет защиты и функция `issueTokens`, начисляющая токены акционерам из списка.



4.2 Обзор уязвимостей

`arbitrage.sol`
`Ownable.sol`
2:

Контракт имеет функцию смены владельца, так как использует в своей основе стандартный контракт Ownable.sol, который реализует передачу прав на другой Ethereum-аккаунт с помощью функции `transferOwnership`.

Текущий владелец может добавить фиктивного акционера в реестр, начислить ему токены, а затем сделать его новым владельцем контракта. Таким образом, **владельцу удастся обойти запрет на добавление себя в реестр**.

В случае, если распределение финансов состоится, то владельцу удастся и вывести средства на тот же фиктивный адрес, так как функции перевода не имеют должных проверок.

5. Заключение. Рекомендации:

Рекомендуем пересмотреть проверочные условия в функциях добавления акционеров и выпуска токенов, также, возможно, добавить условия, не допускающие выплат на кошелёк владельца.

Рекомендуем избавиться от возможности смены владельца аккаунта, или переписать функцию таким образом, чтобы новым владельцем мог стать только тот, у кого нет токенов.

В случае, если из двух пунктов выше будет реализован один, появится риск сценария «блокировки средств»: смена владельца будет происходить злонамеренно, например, на адрес мажоритарного акционера, который, не передав владение кому-либо дальше, не сможет получить свои средства (так как ему, ставшему новым владельцем контракта, это будет запрещено).

Также рекомендуем обратить внимание на раздел «Общие рекомендации» выше.

D-D-S для IFT Token
Москва 2018