AceBusters economy contracts audit by Ambisafe Inc.
August, 2017
Andrii Anisimov

1. **Introduction.** Acebusters requested that Ambisafe perform an audit of the smart contracts implementing their economy concept. The contracts in question are hosted at:

   https://github.com/acebusters/economy
   Latest commit hash at master branch: d0af6b8dcff4e2240b5f9c2f1609290cbb9b1ebc

2. **Disclaimer.** The audit makes no statements or warranties about utility of the code, safety of the code, suitability of the business model, regulatory regime for the business model, or any other statements about fitness of the contracts to purpose, or their bugfree status. The audit documentation is for discussion purposes only.

3. **Executive Summary.**
   Contracts follow main points of Smart Contract Best Practices including reasonable overflow/underflow protection, Pull over Push withdrawals, correct integer rounding, and a lot of assertions all over the code base. Though, several minor suggestions can be applied:

   - Solidity `**pragma**`s should be locked
   - Public functions can be marked as such with `**public**` keyword
   - No protection against accidental funds sending to 0x0 address
   - Claiming ETH after sell is implicit, no withdraw method provided
   - **PowerEvent.sol** constructor lacks checks
   - **Nutz.sol:143,162,164** : `**assert**`s can be swapped to `**require**`s
   - **Nutz.sol:166** : `**transfer**` can be used instead of `**send**`
   - **PowerEvent.sol:123** : `**uint8**` can be used instead, to avoid possible Block Gas Limit issue

4. **Critical Bugs and Vulnerabilities.** No places in code were identified as critical issues.