LakeDiamond: Token Contract Review

Jean-Philippe Aumasson

15/10/2018

Summary

We reviewed the Ethereum contract at https://gitlab.com/lakediamond/lkd-smartcontract/tree/master, defining the LakeDiamond Token (LKD) as an ERC20 token with 141,120,000 undividable units.

The LKD token is based on the ERC20 contract from OpenZeppelin, "a battle-tested framework of reusable smart contracts", a widely used and reviewed suite of contracts. The ERC20 logic is fairly simple and well understood. The basic ERC20 interface is extended with the "burn" functionality from OpenZeppelin's ERC20Burnable.sol contract, which allows to irreversibly destroy tokens. Again, this functionality is well understood and reviewed, minimizing the risk. The only piece of custom code is therefore the following, which defines the LKD token as a burnable ERC20 token:

```
contract LKDToken is ERC20Burnable {
    string public name = "LakeDiamond Token";
    string public symbol = "LKD";
    uint256 public decimals = 0;

    constructor() public {
        _mint(msg.sender, 141120000);
    }
}
```

We believe that the highest risk are therefore the following:

- Tokens being initially assigned to the contract owner, the seed/private key should be safely stored and backed up, to avoid theft and loss of funds. This should preferably be done through a threshold secret-sharing scheme, to ensure that no single individual has access to the tokens.
- Inherent risks to ERC20 tokens, in particular the misuse of the `transfer()` function.