ORS Token Sale Smart Contracts

SICOS

May 12, 2018 - internal use only

Abstract

Latest changes to contract specifications of the ORS tokensale during audit phase.

The ORS tokensale contains of two smart contracts. The token contract is an standard contract from the Open Zeppelin library. The token sale contract is minting tokens directly to the addresses of the investors. There will be a KYC process done by an KYC service provider with an integrated KYCBase interface. Also an ICOEngineInterface provides informational functions for the Eidoo wallet.

1 Token Contract

The token contract implements a ERC20 standard token. It is named ORS Token. Ticker symbol will be ORS. The number of decimals will be 18 to keep the ORS resolution identical to Ξ.

The token contract emits the standard ERC20 events including a transfer event to address 0x0 in case of issued tokens.

We rely on the broadly trusted Open Zeppelin [1] implementation of an ERC20 compliant Token. The following extensions are used:

1.1 Capped and Mintable

Tokens are minted on demand by the owner of the token contract. Therefore the ownership of the token contract has to be transfered to the token sale contract. The minting of tokens is capped at $833,333,333$.

1.2 Pausable

Transfer of tokens is paused on construction of the token contract. Transfer of tokens is unpaused on finalization of the token sale contract. No transfer of tokens is possible during the ICO.

[1] https://github.com/OpenZeppelin/openzeppelin-solidity/tree/master/contracts/token

1.3 Owned

Minting and pausing functions are restricted to the token contract owner. The ownership of the token contract is transfered to the token sale contract immediately after deploy.

1.4 StandardBurnable

Token can be irreversibly burned (destroyed) by the token holder at any time.

2 Token Sale Contract

The token sale contracts implements the Eidoo ICOEngineInterface [2] to provide the Eidoo app with informations on the ICO status.

The token sale contract provides a function that enables the token contract owner to set the ORS Token price at any time. The price represents the ORS per Ξ rate. With a target price of 0.05 € per ORS we will have a rate of $\approx 13,000$ according to a price of ≈ 650 € per Ξ.

Tokens are issued immediatly within the transaction receiving the Ξ payment.

Tokens are minted from different pools in the token sale contract. Any try to mint tokens from an empty pool will revert the whole transaction.

Pool		Cap	Drained
Presale	P	222,247,844	any time before finalization
Main ICO	M	281,945,791	between start and end date if ICO
Bonus	B	60,266,365	together with M and at finalization
Team	T	83,333,333	at finalization
Company	C	127,206,667	at finalization
Advisors	A	58,333,333	at finalization
		833,333,333	

The token sale consists of the three stages main ICO, presale and finalization:

2.1 Main ICO

The Main ICO starts at 2018-05-14T09:00:00+02:00 Unix timestamp 1526281200 and ends at 2018-05-26T23:00:00+02:00 Unix timestamp 1527368400. There may be an early end if $|M| = 0$.

[2] https://github.com/eidoo/icoengine/blob/master/contracts/ICOEngineInterface.sol

Issued tokens are taken from M. Additional 5 % bonus tokens are taken from B if the KYC is signed by the Eidoo Wallet. Requirement for the sizes of M and B is:

$$|B| \quad \geq \quad \frac{|M|}{20} \tag{1}$$

When the cap of M is reached, the last investor will get the last tokens and the remaining Ξ will be refunded. Remaining tokens in M after the end of the ICO will not be issued at all.

2.2 Presale

At any time before finalization tokens are issued to presale investors by the owner of the token sale contract according to a list provided by ORS. Bonus tokens issued to presale investors are added to the token amount off-chain. Issued Tokens are taken from P. The list contains the amounts of tokens assigned to presale buyer addresses and complies to the choosen cap of P. Let (a_i, p_i) be the presale list entry issuing p_i tokens to address a_i. Requirement for the list with size n is

$$|P| \quad = \quad \sum_{i=1}^{n} p_i \tag{2}$$

because $|P| = 0$ is a precondition for finalization of the token sale contract.

2.3 Finalization

After end of main ICO and completness of presale the finalization stage takes place. $|T| + |C|$ tokens are issued to company wallet. $|A|$ tokens are issued to advisors wallet. Remaining $|B|$ tokens are issued to bounty wallet. Further miniting of tokens in token contract is disabled. Transfers are unpaused in token contract. The ownership of the token contract is not transferred. The token sale contract is useless from now on. The token contract has no owner capable of acting.

2.4 KYC

The token sale contract implements the Eidoo KYCBase [3] (ors-integration branch) contract. Eidoo provides two addresses of KYC signers. The first belongs to Eidoo wallet investors and will provide 5 % bonus tokens. The second address belongs to all investors using Eidoo without the wallet.

[3] https://github.com/eidoo/icoengine/blob/master/contracts/KYCBase.sol

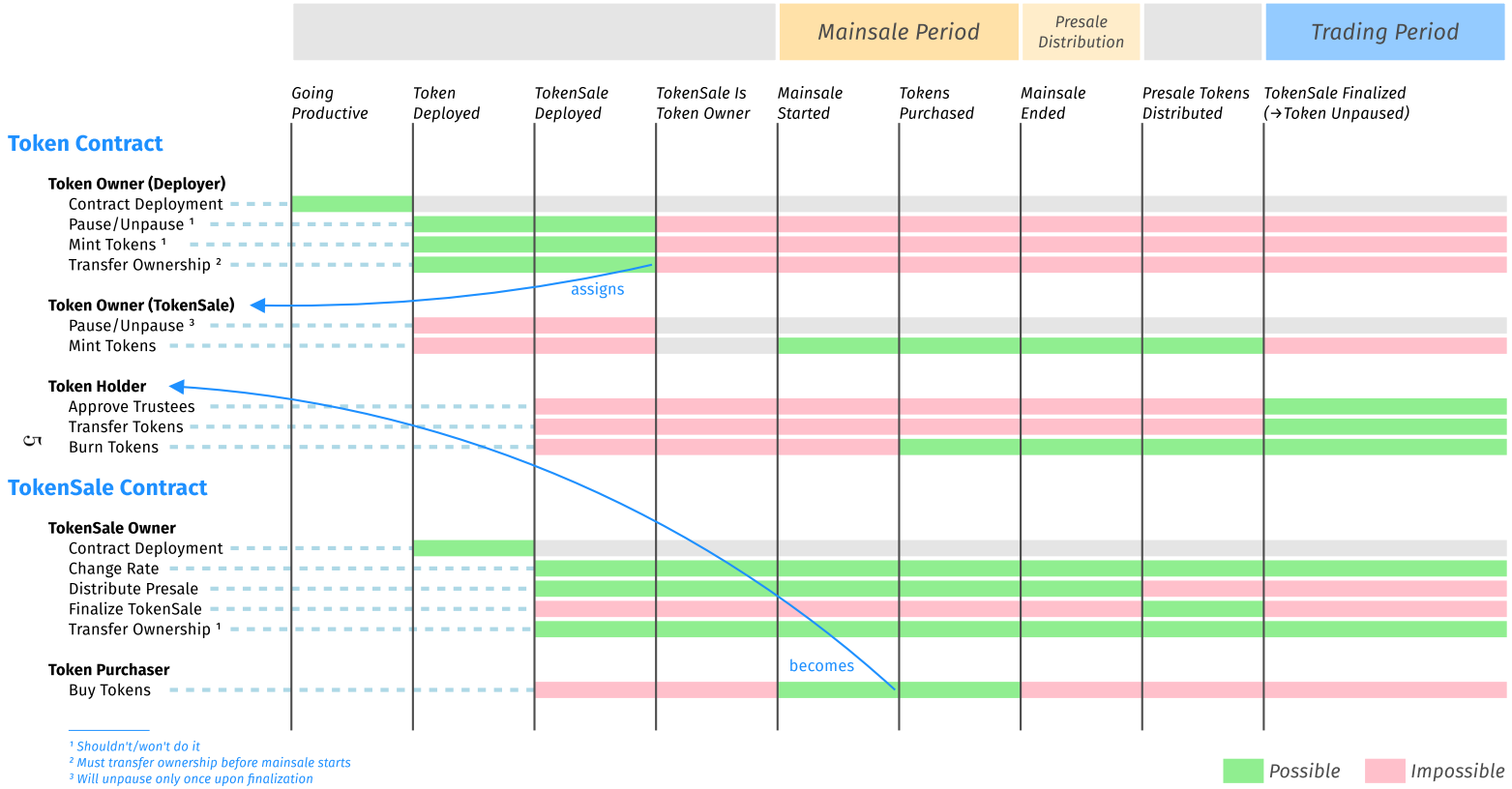
3 Timeline

Date	Event
≈ 2018-05-11	Token contract deployment
	Token sale contract deployment
	Transfer of token contract ownership to token sale contract
	Etherscan code verification
	Transfer of ownership of token sale contract to ORS
	Issuing of presale tokens start
2018-05-14	Main ICO start
<= 2018-05-26	Main ICO end
≈ 2018-05-27	Finalization of token sale contract
	ERC20 transfers enabled

4 Deployment Requirements

The following requirements have to be fulfilled at deployment time 2018-05-11:

Requirement	Source	Value
Size of P and M	ORS	see above
Initial token sale contract owner	SICOS	`0x6aA5a27132f2A828350e06Fb20e04e7E5e205e9A`
Main ICO wallet	ORS	`0xF5A4cD4E15617028188OB645C2c36e4Da5610284`
Bounty wallet	ORS	`0x6265D03b984C3e1378e42B132D2D76F5A1ccb9fF`
Advisors wallet	ORS	`0x43C954E971DB80573861baf0BDfCac1F69f8C0D5`
Company wallet	ORS	`0xF83547c41DBf888DA26d6e3F4A8C0dbA30134672`
Eidoo wallet signer address	Eidoo	`0xdd5ecefcaa0cb5d75f7b72dc9d2ce446d6d00520`
Second signer address	Eidoo	`0x4e315e5de2abbf7b745d9628ee60e4355c0fab86`


Mainsale Period
Presale Distribution
Trading Period
Going Productive
Token Deployed
TokenSale Deployed
TokenSale Is Token Owner
Mainsale Started
Tokens Purchased
Mainsale Ended
Presale Tokens Distributed
TokenSale Finalized (→Token Unpaused)
Token Contract
Token Owner (Deployer)
Contract Deployment
Pause/Unpause ¹
Mint Tokens ¹
Transfer Ownership ²
assigns
Token Owner (TokenSale)
Pause/Unpause ³
Mint Tokens
Token Holder
Approve Trustees
Transfer Tokens
Burn Tokens
TokenSale Contract
TokenSale Owner
Contract Deployment
Change Rate
Distribute Presale
Finalize TokenSale
Transfer Ownership ¹
becomes
Token Purchaser
Buy Tokens
¹ Shouldn't/won't do it
² Must transfer ownership before mainsale starts
³ Will unpause only once upon finalization
Possible
Impossible