

## Second Mineable Token

SMINT



The original text of this document is written in Russian, so in case of inaccuracy in the translation, always look in Russian version of the document.

The purpose of creating this token is to enable Ethereum users to mine tokens, not just buy / transmit. Of course, mining of tokens in the classical sense is impossible - after all, in a second, millions of hashes move through the mining, and only one hash per minute can move there. However, we tried to do something similar to mining.

For the average user, who does not need to understand the programming proficiency, everything is simple - he sends 0 Ether to the contract address and receives from 0 to 500 SMINT tokens + additional reward to his wallet.

Immediately we foresee a lot of questions, we will try to answer them.

**Question: Why should I buy tokens, if I can then just mine them?**

Because the probability of successful mining directly depends on the number of tokens on your balance. To start the mining you need to have at least 0.0001% of all tokens on the balance (the percentage is shown in the online wallet). Even with 100% of tokens, the chance of successful mining is 50%, not 100%. With 0.0001% of tokens, the chance equals 25%. The maximum reward for successful mining also depends on the number of tokens on the balance. Thus, the cost price of the received tokens also depends on the number of unsuccessful attempts (Bad luck).

**Question: How is the reward for mining calculated?**

Answer: The reward for mining is the sum of the basic and additional rewards. The main reward is a random value, the maximum value of which depends on the number of tokens on the balance. The additional reward depends on the main reward and on the time elapsed since the last generation of SMINT tokens (the more time passed, the more reward).

**Question: How do I earn money for SMINT?**

Answer: You can mine tokens and then sell them on the exchange.

**Question: Why would someone buy SMINT tokens on the exchange?**

Answer: Because to start the mining you need to have 0.0001% of all tokens, and SMINT can be used to pay for services.

**Question: How many tokens are assigned to the authors of the contract?**

Answer: 100 billion tokens. These tokens will be sent to the exchange. Part will be spent on all sorts of "bounties."

**Question: How can I know that the contract really fulfills all that is promised?**

Answer: An automatically verified source code for the contract is available on etherscan.io. Thus, you can be sure that the contract works exactly as described in this document.

**Question: How to withdraw frozen tokens?**

Answer: Tokens are thawed automatically during mining.

2 examples:

1. Frozen balance 5 SMINT, mined 2 SMINT. Then the main balance will increase by 4 SMINT (2 + 2), and the frozen one will decrease by 2 SMINT.

2. Frozen balance 1 SMINT, mined 2 SMINT. Then the main balance will increase by 3 SMINT (2 + 1), and frozen becomes 0 SMINT.

**Question: The purpose of this token is only to earn money on it?**

Answer: No, this is a ERC20-token. It can be accepted as payment for services and so on.

**Question: How to pay for services using SMINT?**

Answer: The contract provides convenient functionality for payment of services. Details can be read below.

**Question: Why cannot you limit emission? Will the value of the token increase with unlimited emissions?**

Answer: Emission cannot be limited, because unlike the classical crypto currency, there is no time of block generation in SMINT. This means that one can not foresee what time will be spent for the N minings necessary to reduce the reward. The cost will increase, because with an increase in the total number of tokens, the number of tokens required for the beginning of mining will also increase. This, in turn, will increase the demand for tokens and increase their price. Also, the use of tokens in online stores positively affects the price of tokens.

### Technical details

The most commonly used contract methods:

```
"70a08231": "balanceOf(address)",
"60ea5a3c": "bill(uint256)",
"22b1f39c": "currentInvoice()",
"753ded15": "failsOf(address)",
"be91de53": "frozenBalanceOf(address)",
"4e6d1405": "invoices(uint256)",
"c290d691": "pay(uint256)",
"cd9354e4": "successesOf(address)",
"18160ddd": "totalSupply()",
"a9059cbb": "transfer(address,uint256)",
```

Contract events:

```
"0x4cd36fb76975ab408e67b9ff3c17b4dd03112963aed18ccd8907a0c8b77daec2":
"Mine(address,uint256,uint256)",
"0xc86c16f21f937adc15d9d6ae91cd0c5db78c621ef57872a6486fbab92e325ed8":
"Bill(uint256)",
"0xdddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef":
"Transfer(address,address,uint256)",
```

```
"0x781a2cca10f15083d2de4efa525a850679da338505  
fe558eb29b6ab494738deb":
```

```
"Pay(uint256)",
```

For a mining attempt send 0 Ether to the contract address. In the case of a successful attempt, from 0.1 to 500 tokens + additional reward will be credited to the balance of the miner.

For invoicing, the method `bill(uint256 amount)` is used. The "amount" parameter is the amount in wei. The transaction generates event `Bill(uint256 invoiceId)`, where "invoiceId" is the invoice number.

This number can be transferred to another user to pay the bill. The invoice is paid using the method `pay(uint256 invoiceId)`. If there are sufficient funds on the balance of the user, the payment amount is transferred to the seller, and event `Pay(uint256 invoiceId)` is generated, which can be read by the seller.