# **The Dagger crypto currency: white paper and user guide**

Version 0.2. January 19, 2018

## **1. Introduction**

### **What is it**

Dagger (token XDAG) is a new crypto currency, which is not based on the blockchain, but on an directed acyclic graph (DAG) instead and, unlike other DAG-oriented coins, allows mining.

### **Goal**

The goal of the project is to create a decentralized payment system that allows processing thousands of transactions per second.

### **Ideas**

Each block contains exactly one transaction. At the same time, the block is an address. Among all transactions, the main chain is allocated - it is a chain with the maximum difficulty. In the main chain, new coins are created about once a minute.

### **Roadmap**

Main network was launched January 5, 2018. ICO is not planned. There is no pre-mine. Everyone can participate in mining on equal terms. Only CPU can be used for mining at the present.

**Security**

The ECDSA algorithm with a 256-bit private key is used for a signature that confirms the rights of the wallet owner for money in the given address. All messages are transmitted between hosts in an encrypted form using the author's semi-symmetric encryption algorithm. The session key to it is transmitted using the 8192-bit key RSA algorithm.

### **Team**

The developer of the dagger is an anonymous author under the pseudonym Daniel Cheatoshin (cheatoshin@mail.com). Several people volunteered to help the project.

### **What to see next**

The main site of the project: [http://xdag.me](http://xdag.me/)

Repository for Linux users:<https://github.com/cheatoshin/xdag>

Discuss on the forum:<https://bitcointalk.org/index.php?topic=2552368>

You can also read the further guide to using the command-line full node program.

## **2. Download and install**

**For Windows**

1. Go to Downloads section of the official site<http://xdag.me/downloads.html>
2. Download zip archive:
3. 1. win64exe.zip for 64-bit Windows (x64);
   3. win32exe.zip for 32-bit Windows (x86);
   5. winsrc.rar if you want to build the software from source using MS Visual Studio.

3. Unpack the archive into any folder.

### **For Linux**

1. Install dependencies:

for Debian/Ubuntu:

$ **sudo apt-get install git gcc libssl-dev**

for RedHat/Fedora:

$ **sudo dnf install git gcc openssl-devel**

2. Download source:

$ **git clone****<https://github.com/cheatoshin/xdag.git>**

3. Make the program:

$ **cd xdag/cheatcoin**

$ **make**

## **3. Run the program**

### **For Windows**

1. Open Windows terminal

1. Go to the folder where the downloaded archive is unpacked:

c:\Users\yourname> **cd win64exe**

1. Run the program:

c:\Users\yourname\win64exe> **xdag.exe –d**

### **For Linux**

1. Run the program in daemon mode:

$ **./xdag -d**

1. Connect an interactive terminal to the daemon:

$ **./xdag -i**

### **First-time launch**

After the first start of the program, you should enter a random character sequence.

Enter random phrase: [type characters and press Enter]

#### ***Generating host keys…***

The entered characters will be used to generate random keys. After that, the program will create the file dnet\_key.dat with a random key for the RSA algorithm of the transport layer within a few minutes. Also, the file dnet\_key.dat will be used to generate random data after each subsequent start of the program. The program will also create the wallet.dat file with private keys for the ECDSA signature algorithm. Files dnet\_key.dat and wallet.dat should be kept in secret.

Program will not run if the file netdb.txt is absent in the working directory of the program. In this case the file netdb.txt should be copied from the distribution to the working directory.

### **Flags of the program**

The main flags that one can add to the command line when you run the program:

-h see full list of flags supported by the program;

-m N use N mining CPU threads; it has a sense to start mining after synchronization with the network, see section 4. Synchronization;

-p ip:port provide white ip address of this host and an opened port; other hosts can do an external connections using these ip address and port;

-t connect to test network instead of main network; the only test network is running before December 24, 2017;

For example, if you want to run the program with mining on two CPU cores, in the main net and with external ip address 1.2.3.4 and opened port 13654, run the program with the following flags:

Windows: c:\Users\yourname\win64exe> **xdag.exe -d -m 2 –p 1.2.3.4:13654**

Linux: $ **./xdag -d -m 2 –p 1.2.3.4:13654**

## **4. Working with the program**

### **Interactive mode**

Work with the program is interactive. After starting the program, you are prompted to enter commands. The prompt has the following form:

xdag>

Before the fully functional work with the program, one need to make sure that your host is connected to the dagger network and that you are synchronized with the network.

### **Connection to the network**

To verify that you are connected to the network, you need to enter the following command:

xdag> **net hosts**

The output of this command contains the list of all detected hosts in the dagger network, for example:

Active hosts:

0. 5E546C24 0 sec, T13.753, local trust 127.0.0.1:0

1. E3153759 0 sec, T13.753, immediate trust 46.34.155.102:41962

[etc]

Your host is listed in the first line, the other hosts listed below. You need to make sure that your host is not the only one on the network, there are several other hosts. If there are no other hosts on the network, you need to wait a while and then re-enter the command ‘net hosts’.

### **Synchronization with the network**

If your host is already connected to the network, then to verify that you are synchronized with the network, you need to enter the following command:

xdag> **stats**

The output of this command has the following form:

Statistics for ours and maximum known parameters:

hosts: 176 of 176

blocks: 201816 of 201816

main blocks: 19320 of 19322

orphan blocks: 15

wait sync blocks: 0

chain difficulty: 1439f6f34516c7b2a536 of 1439f6f34516c7b2a536

XDAG supply: 19783680.000000000 of 19785728.000000000

Your host is synchronized with the network if in the line 'chain difficulty' two hexadecimal numbers coincide, for example:

chain difficulty: **1439f6f34516c7b2a536** of **1439f6f34516c7b2a536**

If you are not synchronized with the network, you need to wait a while and then try typing the ‘stats’ command again. Synchronization after the first start of the program can take from several hours to several days depending of the time passed from the start of network.

**Important! Dagger is a real-time payment system. Every minute a new main block is generated. So make sure that the system clock goes correct with the precision up to the few seconds. Also, make sure that the time zone is set correctly. Otherwise, the synchronization may not complete. It is recommended to always enable the NTP service on the host.**

### **Manage your money**

After the synchronization with the network is completed, one can start to manage your account. If you have mining enabled, the balance on your account is automatically increased from time to time.

To view the current balance of your account, run the following command:

xdag> **balance**

To see a list of your own addresses so that other people can transfer money to you, run the command:

xdag> **account**

The following is an example of output of the ‘account’ command:

5ksy3XNRkaS3U9kkpWKvde47Xjij1qSA 100000.000000000 key 0

TpkclaMNUhZSooO9OisrqIzsXbfc4yhz 1024.000000000 key 0

[etc]

Your addresses are: **5ksy3XNRkaS3U9kkpWKvde47Xjij1qSA** **TpkclaMNUhZSooO9OisrqIzsXbfc4yhz** and so on.

To transfer money from your account to another address, run the command:

xdag> **xfer [amount] [address]**

For example:

xdag> **xfer 1000.0 3InPziRPwCzWbVY6c0owfuE5VsfbCZCD**

To see a list of all the commands supported by the program xdag, run the command

xdag> **help**

## **5. Specification**

The full specification of the Dagger protocol one can find in the following document located in the official site:<http://xdag.me/xdag_protocol.doc>