Statistical model checking for blockchain-based applications

D B Maksimov, I A Yakimov and A S Kuznetsov
Siberian Federal University, 79 Svobodny avenue, Krasnoyarsk, 660041, Russian Federation

E-mail dmaksimov@sfu-kras.ru

Abstract. Blockchain-based applications have become increasingly popular in recent years. Security concerns are rising as blockchain users are becoming aware of how their private data is stored across this decentralized network of computing nodes. Formal verification allows developers to improve software quality and hence make it more secure. The main hypothesis of this paper is that it is achievable to perform formal verification of a model that represents a smart contract responsible for user registration. In order to test the hypothesis, we have created a model of such a smart contract and checked it against a set of formal constraints. As a result of the conducted study, a conclusion can be made that the hypothesis is correct.

Nowadays the global Internet network provides users with a wide range of services. On the other hand, ever-growing security risks put higher expectations on software quality. Blockchain technology addresses this issue in its own unique way. It protects users from cybercrimes through decentralization. Each node of a blockchain network stores an entire history of all the transactions ever fulfilled making it really hard for attackers to mess with user assets [1].

At present, the blockchain systems, in addition to the financial sector, in which they are actively used in cryptocurrencies, are used in many areas of human life. For example, in medicine, software is already actively used that allows you to work with patient data from anywhere in the world. It does not depend on the work of a single centralized server. Blockchain technology is also already being used to control the national borders of the Netherlands. Passenger data and metrics are stored securely in a distributed system. In addition to public use, the technology of working in the environment of blockchain is used by large companies, such as Amazon. This company uses the advantages of blockchain technology to improve database security.

Smart contracts play a crucial role in modern blockchain applications. A smart contract is a program that allows for the automatic closing of deals between two or more parties involved. Each smart contract acts as an independent intermediary between the parties while performing transactions. This allows us to avoid possible conflicts of interest and secure user assets.

Strict requirements on smart contracts enforce developers to seek special approaches for assessing their quality. One of the most reliable methods for proving that a program meets a set of requirements is formal verification. Formal verification is an umbrella term covering a wide range of methods [2], in particular, model checking.

Most of the methods of the model checking represent the target program in the form of an automaton. The main goal is to check how much the program conforms to a set of formal constraints expressed in the language of linear temporal logic [3].

Model-checking is performed during the following 3 stages [4] as shown in figure 1:
In this paper, the process of user registration in the system is simulated. During registration, a short random unique name (aka alias) is generated for each user. However, the number of aliases that the system is able to generate is limited by the storage capacity. Relatively low speed of operations in blockchain and costs associated with each individual transaction affect the overall user experience and the ability to get an alias on time.

The diagram shown in figure 2 represents all the possible states and transitions of the simulated system as an oriented graph where:

a) $A_0$ – initial state;

b) $A_1$ – registration trial;

c) $A_2$ – successful registration;

d) $A_3$ – registration failure.
The set of final states of the simulated system includes A₂ and A₃. Each of these states signals the successful termination of the user registration process with either positive or negative results. For each transition to happen, the following conditions have to be satisfied:

- T₁ – is only allowed, when the maximum number of user attempts to get an alias has not yet been reached;
- possibility of T₃ depends on the capacity of the storage – the more aliases are already in the storage, the less likely the user is to get her alias;
- T₂ is possible only if the user has reached the maximum number of attempts to get an alias;
- T₄ is a technical transition necessary for the correct system shutdown after the system has reached one of the final states.

![Diagram of the simulated system.](image)

The main requirement for the system described in this work is the following: during registration, the user can get an alias or a rejection after a number of attempts not exceeding x. This requirement is expressed as a linear temporal formula in the following way:

\[
\phi(t) \equiv \Diamond_{[0, \] \} \left( \neg (\text{System.state} = 0) \land (\text{System.c} \leq x) \right)
\]  

(1)

The parametric study in this paper is based on the method described in [1]. The parametric study is understood as an automated way of checking the models on the family of properties expressed in the form of function φ(x), where x is an integer parameter, for which the initial and final values, as well as the step of change, are given.

To evaluate the results of model checking a method of statistical analysis of stochastic systems was used, specifically the method of probability estimation - PESTIM (Probability Estimation). As a result of its work, this method should give an answer to the question: "What is a probability P that system S satisfies a given set of constraints?"
One of the parameters of the PESTIM method is accuracy $\delta$ which shows that the procedure computes such $p'$ that $|p' - p| \leq \delta$ with confidence $1 - \alpha$. The method is based on the Chernoff-Hoeffding bound [5].

The study was conducted in an integrated SBIP development environment that provides a visual interface for working with the «Behavior Interaction Priority» framework – BIP (Behaviour Interaction Priority) – software for modeling stochastic systems. It’s freely available at http://www-verimag.imag.fr/Statistical-Model-Checking.html.

A statistical modeling module is connected to this framework, a tool that allows simulation of the simulated system, while simultaneously tracking system parameters during the simulation. It also connects a module for parameter studies, which allows you to calculate parametric properties during the system operation.

During the study, measurements of the probability of the user passing the registration process on a number of attempts were made. On each iteration of the study, the storage attribute varied from 0.1 to 1.0. The study was conducted with parameters $\alpha = 0.1$ and $\delta = 0.1$.

The results of the study as shown in figure 3 are as follows:

- The more the storage is filled up, the more time the user needs to go through the registration process;
- After a certain number of attempts to obtain an alias defined by the N parameter, a user is guaranteed to be rejected in registration.

In the course of the study, the following hypothesis was proposed and experimentally proved: "a model of a smart contract for registration of new users can be created and formally verified". The model and the system of constraints on the language of linear temporal logic is created. Formal verification has allowed us to estimate quantitatively the degree of compliance of the built model to the system of constraints imposed on it.

![Research of the user registration process](image-url)

**Figure 3.** Survey results.

As a result of this work, a study with the parameters $\alpha = 0.1$ and $\delta = 0.1$ was carried out, and the hypothesis under consideration turned out to be true at these parameters.
In the future, it is planned to expand the research model and add a malefactor to the investigated model in order to analyze the possibility of obtaining someone else's user alias during the registration process. It is also planned to use the method of hypothesis testing to analyze the data obtained during the study.

References
[1] 2017 Decree of the Government of Russian Federation 28.07.2017 1632-r
[2] Barinova A A and Zapechnikov S V 2017 On the Techniques and Tools for Privacy-Preserving Smart Contracts IT Security 24(2) 16–23
[3] Rozier K Y 2011 Linear Temporal Logic Symbolic Model Checking Comp. Sc. Rev. 5(2) 163–203
[4] Baier C and Katoen J P 2008 Principles of Model Checking (Cambridge, London: The MIT Press) p 984
[5] Hoeffding W 1963 Probability Inequalities for Sums of Bounded Random Variables J. Amer. Stat. Assoc. 58(301) 13–30