The Smart Contract Guard Model

Thaleia-Elpis Kavalierou
Department of Informatics and Computer Engineering
University of West Attica
Athens, Attica, Greece
cs171009@uniwa.gr

Ioanna Kantzavelou
Department of Informatics and Computer Engineering
University of West Attica
Athens, Attica, Greece
ikantz@uniwa.gr

ABSTRACT
Blockchain technology is extensively used where distributed and decentralized services are required. This paper proposes the Smart Contract Guard (SCG) model, with a blockchain approach that aims to shield the state of smart contracts from malicious actors and protect their storage.

KEYWORDS
Blockchain, Ethereum, smart contracts, guard, SCG model.

ACM Reference format:

1 Introduction
A blockchain system allows transactions without any intervention of a trusted third party. Smart Contracts (SC) are programs that run when certain predefined conditions are met, and consist of transactions processed in distributed blockchains [1]. Among their discovered vulnerabilities, we identified some that target the storage of SC [2]. However, a vulnerable SC, already deployed in the Ethereum blockchain [1], cannot be modified in order to fix the vulnerability. The proposed Smart Contract Guard (SCG) model uses a sandbox, which runs the SC, regardless its vulnerabilities, using sandbox’s storage. The main contribution of the proposed work is a method, incorporated in the SCG model, which preserves the real SC storage, by setting another SC in front of the real SC and combining a sandbox and the blockchain technology.

2 The Smart Contract Guard (SCG) Model
The SCG model employs a sandbox, a virtual environment for executable code testing used in computer security. Unlike other solutions, a sandbox does not control the code of the Real SC, but executes it, despite its vulnerabilities, in its own storage. Any “bad developments” will be detected during the execution time.

A Sandbox SC could protect the storage of the Real, which guards, by putting its own in the “front line”. Following this pattern, the Sandbox SC storage will have the same structure as the Real. The only difference is that it will not hold real data but it will be informed about the data current state by communicating with the Real SC.

The SCG model, presented in Figure 1, illustrates how it would protect a SC from a malicious actor. The Guard is the sandbox and the Real SC is the SC. They communicate by exchanging information regarding Real’s data state (data, cryptocoins), the Sandbox SC, and the Real’s method. An external user can take advantage only of Real’s functionality, by passing through the Sandbox SC. The red arrow indicates that a potential malicious actor cannot directly contact to the Real SC.

![Figure 1: The Smart Contract Guard model.](image)

3 Conclusion
Every exploit could have been avoided by securing the access to SC’s storage. The SCG model demonstrates how this condition can be resolved with the use of a Sandbox SC that seeks for security risks at runtime. Comparing to other approaches, with the most popular of them focusing on code auditing, this model aims to stand as a guard between the application’s logic and the communication with its outside world. The model can be applied wherever an additional security level is required.

REFERENCES