Towards benchmarking of Solidity verification tools

Massimo Bartoletti
University of Cagliari, Italy

Fabio Fioravanti
University of Chieti-Pescara, Italy

Giulia Matricardi
University of Chieti-Pescara, Italy

Roberto Pettinau
Technical University of Denmark, Denmark

Franco Sainas
EPFL, Switzerland

Abstract

Formal verification of smart contracts has become a hot topic in academic and industrial research, given the growing value of assets managed by decentralized applications and the consequent incentive for adversaries to tamper with them. Most of the current research on the verification of contracts revolves around Solidity, the main high-level language supported by Ethereum and other leading blockchains. Although bug detection tools for Solidity have been proliferating almost since the inception of Ethereum, only in the last few years we have seen verification tools capable of proving that a contract respects some desirable properties. An open issue is how to evaluate and compare the effectiveness of these tools: indeed, the existing benchmarks for general-purpose programming languages cannot be adapted to Solidity, given substantial differences in the programming model and in the desirable properties. We address this problem by proposing an open benchmark for Solidity verification tools. By exploiting our benchmark, we compare two leading tools, SolCMC and Certora, discussing their completeness, soundness and expressiveness limitations.

2012 ACM Subject Classification Software and its engineering → Formal software verification

Keywords and phrases Smart contracts, Ethereum, Verification, Blockchain

Digital Object Identifier 10.4230/OASIcs.FMBC.2024.6

Supplementary Material https://github.com/fsainas/contracts-verification-benchmark

Funding *Massimo Bartoletti*: Partially supported by project SERICS (PE00000014) and PRIN 2022 DeLiCE (F53D23009130001) under the MUR National Recovery and Resilience Plan funded by the European Union – NextGenerationEU.

1 Introduction

The rapid growth of decentralized applications based on blockchain technologies have emphasized the importance of ensuring the security of smart contracts — the basic building blocks of these applications. The research on smart contracts security has been proliferating since 2016, leading on the one side to the discovery of a variety of attacks, and on the other side to the development of several tools to detect vulnerabilities of smart contracts before they are deployed. Despite the increasing breadth and precision of these analysis tools, attacks to smart contracts have caused financial losses worth several billions of dollars so far, and are unlikely to be eradicated anytime soon.

A large class of analysis tools for smart contracts are focussed on detecting known vulnerability patterns in contracts code. Even though tools of this type can detect many nefarious bugs, statistically the vast majority of the losses due to real-world attacks are

caused by logic errors in the contract code, which cannot be prevented by only checking for fixed vulnerability patterns [5]. In this context, a contract can be considered secure when its executions are coherent with some ideal behaviour, even in the presence of adversaries trying to subvert it. Only a few tools support this kind of security analysis, allowing developers to specify the ideal properties the contract is expected to satisfy. In this work we focus on SolCMC and Certora, two leading verification tools for contracts written in Solidity, the main smart contract language for Ethereum and EVM-compatible blockchains. Both tools allow the developer to specify desirable contract properties, and use SMT solvers to verify whether the contract satisfies them, showing a counterexample when detecting a violation. Although both tools have been independently tested by their developers [1, 4], no public comparison exists so far to assess their effectiveness and limitations in practice.

Our long-term goal is a comprehensive, publicly available benchmark to evaluate the effectiveness of verification tools for Solidity contracts. As an initial step towards this goal, in this paper we present a benchmark comprising 323 verification tasks, each one made of a Solidity contract and a property it is expected to satisfy.[1] A crucial component of our benchmark is a manually crafted ground truth of the verification tasks, encompassing multiple versions of each smart contract in order to cover different ways of satisfying or violating its associated properties. To foster the reproducibility of the results, we make available a toolchain that automatises the construction of the verification tasks, their processing with SolCMC and Certora, and the summarisation of the results. Based on these artifacts, we present a preliminary evaluation of SolCMC and Certora, comparing their completeness, soundness, and expressiveness. Finally, we have introduced a scoring scheme for Solidity verification tools [2], which is inspired by schemes used in software verification competitions [3], but taking into account the peculiarities of the smart contracts context.

References

1 Leonardo Alt, Martin Blicha, Antti E. J. Hyvärinen, and Natasha Sharygina. Solcmc: Cav 2022 artifact. https://github.com/leonardoalt/cav_2022_artifact/tree/main, 2022.

2 Massimo Bartoletti, Fabio Fioravanti, Giulia Matricardi, Roberto Pettinau, and Franco Sainas. Towards benchmarking of solidity verification tools. In *5th International Workshop on Formal Methods for Blockchains (FMBC)*, 2024. To appear.

3 Dirk Beyer. Competition on software verification and witness validation: SV-COMP 2023. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, volume 13994 of *LNCS*, pages 495–522. Springer, 2023. doi:10.1007/978-3-031-30820-8_29.

4 Certora. Formal verification of OpenZeppelin (may-june 2022). https://github.com/OpenZeppelin/openzeppelin-contracts/blob/master/certora/reports/2022-05.pdf, 2022.

5 Stefanos Chaliasos, Marcos Antonios Charalambous, Liyi Zhou, Rafaila Galanopoulou, Arthur Gervais, Dimitris Mitropoulos, and Ben Livshits. Smart contract and DeFi security: Insights from tool evaluations and practitioner surveys. In *ICSE*, 2024. To appear.

[1] https://github.com/fsainas/contracts-verification-benchmark