# Article information:

WASAI: uncovering vulnerabilities in Wasm smart contracts | Proceedings of the 31st ACM SIGSOFT International Symposium on Software Testing and Analysis  
<https://dl.acm.org/doi/abs/10.1145/3533767.3534218>

# Article summary:

1. WebAssembly (Wasm) smart contracts are vulnerable to various attacks, but few developers release the source code for security review.

2. WASAI is a new concolic fuzzer designed to uncover vulnerabilities in Wasm smart contracts.

3. Experiments show that WASAI outperforms existing methods and can detect over 70% of vulnerable contracts on the EOSIO Mainnet.

# Article rating:

Appears moderately imbalanced: The article provides some useful information, but is missing several important points or pieces of evidence that would be required to present the discussed topics in a balanced and reliable way. You are encouraged to seek a more balanced perspective on the presented issues by exploring the provided research topics and looking at different information sources.

# Article analysis:

The article provides an overview of WASAI, a new concolic fuzzer for uncovering vulnerabilities in Wasm smart contracts, and presents the results of experiments conducted to evaluate its performance. The article is well-written and provides evidence for its claims, such as the F1-measure of 99.2%, which demonstrates that WASAI outperforms existing methods. However, there are some potential biases and missing points of consideration that should be noted when evaluating the trustworthiness and reliability of this article.

First, the article does not provide any information about the authors or their affiliations, which could lead to potential bias in favor of their own work or research interests. Additionally, while the article mentions that over 300 vulnerable contracts have not been patched yet, it does not provide any information about how these vulnerabilities were addressed or what measures were taken to ensure they do not occur again in future deployments. This lack of information could lead to a one-sided reporting bias in favor of WASAI's effectiveness without considering possible risks associated with its use or other counterarguments against it.

Finally, while the article does mention some limitations associated with existing approaches for detecting vulnerable Wasm smart contracts (e.g., low code coverage), it does not explore any unexplored counterarguments or present both sides equally when discussing these limitations and how they compare to WASAI's performance. This partiality could lead readers to form an incomplete understanding of WASAI's capabilities without considering all relevant points of view on this topic.

# Topics for further research:

* Wasm smart contract vulnerabilities
* Risk assessment of Wasm smart contracts
* Counterarguments against WASAI
* Code coverage of Wasm smart contracts
* Patch management of Wasm smart contracts
* Limitations of existing Wasm smart contract fuzzers

# Report location:

<https://www.fullpicture.app/item/c07856b7b6f856e18e5196001594fc1a>