Architecture

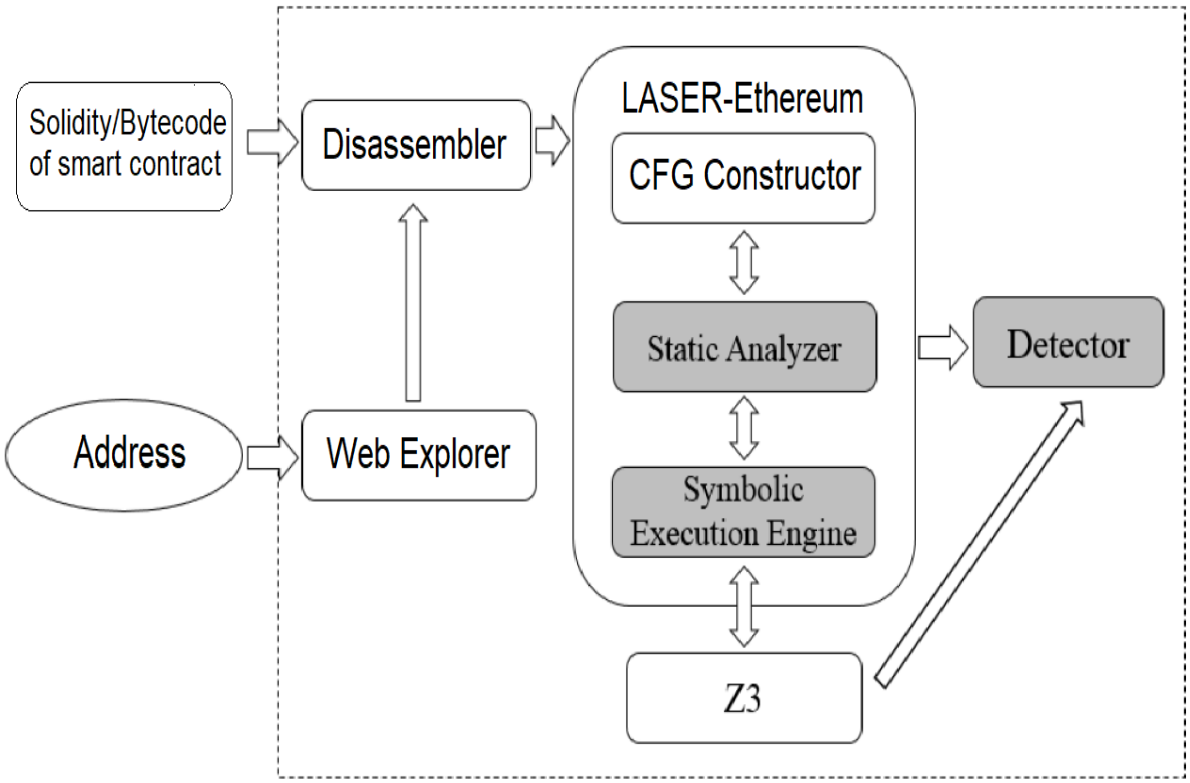

Figure 1: Architecture of the vulnerability detection system

Methodologies/Algorithms

1. **Target-Guided Automated Testing Algorithm:** This algorithm is used to cover all critical paths as quickly as possible in automated testing.
2. **Unrelated Path Pruning Algorithm:** This algorithm is used for removing of unrelated paths. If a path is likely to reach a critical node that has not been covered in the tested contract, then it is the relevant path, otherwise, it is an unrelated path.
3. Algorithms that show the vulnerability detection logic.

Applications

1. Improves code quality.
2. Detect payment bugs in smart contracts and prevent financial losses.
3. Prevent jeopardy of digital assets.
4. Ensures safe use of smart contracts for transfer of digital assets.

References

1. Luu, L., Chu, D.-H., Olickel, H., Saxena, P., & Hobor, A. Making Smart Contracts Smarter. Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. 2016
2. Karthikeyan Bhargavan, Antoine Delignat-Lavaud, Cédric Fournet, Anitha Gollamudi, Georges Gonthier, Nadim Kobeissi, Natalia Kulatova, Aseem Rastogi, Thomas Sibut-Pinote, Nikhil Swamy, and Santiago Zanella-Béguelin. "Formal Verification of Smart Contracts: Short Paper". In Proceedings of the 2016 ACM Workshop on Programming Languages and Analysis for Security (PLAS '16). ACM, New York, NY, USA, 2016.
3. Chen, T., Li, X., Luo, X., & Zhang, X. (2017). "Under-optimized smart contracts devour your money." 2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER).2017
4. Wohrer, M., & Zdun, U . "Smart contracts: security patterns in the Ethereum ecosystem and solidity.". 2018 International Workshop on Blockchain Oriented Software Engineering (IWBOSE),2018
5. Bo Jiang, Ye Liu, and W.K. Chan. "ContractFuzzer: Fuzzing Smart Contracts for Vulnerability Detection." In Proceedings of the 33rd IEEE/ACM International Conference on Automated Software Engineering, 2018.
6. P. Tsankov, A. Dan, D. Drachsler-Cohen, A. Gervais, F. Buenzli, and M. Vechev. "Securify: Practical security analysis of smart contracts", presented at the 25th ACM Conf. CCS, Toronto, ON, Canada. Oct. 2018.
7. S. Kalra, S. Goel, M. Dhawan, and S. Sharma. "Zeus: Analysing safety of smart contracts", in Network and Distributed Systems Security(NDSS) Symposium, 2018
8. Nikolić, Ivica, Aashish Kolluri, Ilya Sergey, Prateek Saxena, and Aquinas Hobor. "Finding the greedy, prodigal, and suicidal contracts at scale." In Proceedings of the 34th Annual Computer Security Applications Conference, pp. 653-663. ACM, 2018.
9. Albert, E., Correas, J., Gordillo, P., Román-Díez, G., & Rubio, A. "*SAFEVM*: a safety verifier for Ethereum smart contracts." Proceedings of the 28th ACM SIGSOFT International Symposium on Software Testing and Analysis .2019.