Projeto de Formatura – 2023

PCS - Departamento de Engenharia de Computação e Sistemas Digitais

Engenharia Elétrica – Ênfase Computação

Tema: **Building a Database for Evaluating Smart Contract Vulnerabilities Detection Tools**

Resumo

Com o aparecimento das criptomoedas, transações em ambientes de blockchain, como a rede Ethereum, se tornaram uma prática comum. Conforme o volume de transações cresce, novas ameaças nos chamados contratos inteligentes emergem continuamente. Procurando evitar problemas nas transações, uma série de ferramentas de detecção de vulnerabilidades foram desenvolvidas ao longo do tempo. Contudo, do conhecimento do autor, ainda não existe uma base de dados com contratos construídos especificamente para validação dessas ferramentas. O resultado final deste estudo foi a implementação de tal base, que seja o mais representativa possível das vulnerabilidades presentes no mundo real na rede Ethereum.

Motivação

Com o aumento do volume de transações de criptomoedas, houve também um aumento significativo na exploração de vulnerabilidades nos respectivos blockchains a fim de se obter valores indevidamente. No caso da rede Ethereum a exploração se deu por meio de vulnerabilidades nos contratos inteligentes (SCs) da rede, que são pequenos códigos escritos e publicados pelos usuários. Um dos mais notáveis ataques foi apelidado de "TheDAO", em 2016, no qual foram desviados valores somando mais de US$320.000.000,00.

Os crescentes ataques aos contratos inteligentes da rede Ethereum motivaram a comunidade científica e usuária a desenvolver ferramentas de detecção de vulnerabilidades nos códigos dos contratos inteligentes, na busca de um ambiente de negociação em blockchain mais seguro. Entretanto, indaga-se: Como estas ferramentas tem sido validadas quanto às suas respectivas capacidades?

Estado da Arte

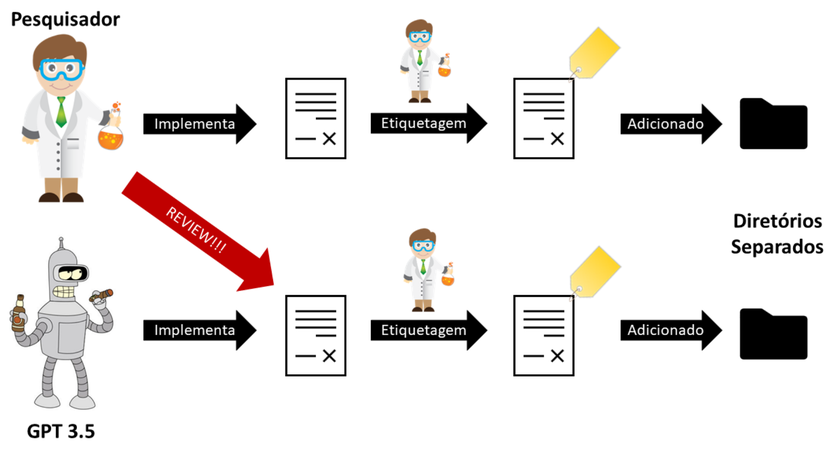
Como método de validação para ferramentas de detecção de vulnerabilidades em SCs, foram criadas uma série de bases de dados contendo SCs com vulnerabilidades seguindo duas técnicas principais:

- Uma série de SCs foram extraídos da rede do Ethereum e rotulados manualmente ou por um compilado de ferramentas de detecção presentes na literatura [1]
- Uma série de SCs foram implementados especificamente para conter vulnerabilidades conhecidas e rotuladas com alto grau de assertividade [2]

Entretanto, os autores de [3] e [4] notaram a inexistência de uma base de dados que contenham, ao mesmo tempo, rótulos assertivos e SCs que realmente sejam representativos de casos de uso reais sem depender de outras ferramentas de detecção da literatura que, provavelmente, também foram validadas desta maneira.

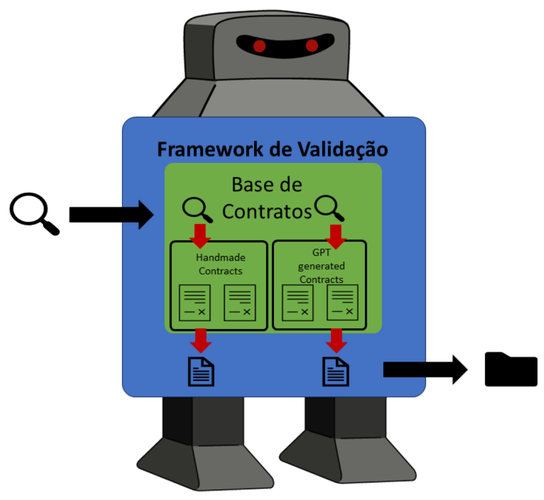
Descrição do Projeto

Este projeto implementa SCs escritos em Solidity manualmente e com auxílio de IA generativa do tipo GPT 3.5, rotulados manualmente ao longo de sua implementação. Estes SCs foram desenvolvidos utilizando-se do RemixIDE e testados em uma rede blockchain local de teste, construída por meio do Ganache. Por fim, um script em *bash* permite que os contratos sejam varridos automaticamente pela ferramenta que se deseja validar.



Objetivo

Este projeto se propõe a apresentar uma base de dados composta por uma série de SCs que implementem funcionalidades reais presentes nos SCs na rede Ethereum, na forma de um *playground* feito especificamente para a validação de ferramentas de detecção de vulnerabilidades. A base de dados pode ser varrida automaticamente utilizando um script *bash* compatível com qualquer ferramenta de detecção de vulnerabilidades.



Metodologia

Primeiramente, foi desenvolvido um contrato inteligente cuja única função era implementar a vulnerabilidade. Este contrato foi utilizado para a compreensão do funcionamento de cada uma das vulnerabilidades. Depois, foram implementados dois contratos, um mais simples e outro mais complexo, que apresentassem funcionalidades reais de contratos inteligente do Ethereum e apresentassem uma das vulnerabilidades. Por fim, foi utilizado o GPT 3.5 para gerar mais dois contratos com a mesma premissa dos dois anteriores. Assim, foi somado ao conhecimento do autor o próprio treinamento da ferramenta de AI para que se obtenha uma base de dados representativa de aplicações reais.

Resultados

Como resultado obteve-se uma base de dados contendo 50 SCs que implementam 10 vulnerabilidades conhecidas já descritas na literatura. Cada vulnerabilidade apresenta aproximadamente 5 exemplos, sendo 3 desenvolvidos manualmente com complexidade crescente (ou seja, um SC contendo apenas a vulnerabilidade, outro apresentando uma aplicação mais complexa e um último contendo uma implementação real que apresente a vulnerabilidade), bem como 2 SCs implementados automaticamente por meio do GPT 3.5.

Para garantir que a base de contratos inteligentes está implementada corretamente foram executadas contra a base duas ferramentas de detecção, uma estática e outra dinâmica. Ambas foram capazes de varrer todos os contratos da base e fornecer relatórios não vazios de resultado. Nota-se que, em [3], os autores notam que nenhuma da ferramentas de detecção testadas foi capaz de varrer mais de 50% da base de contratos montada.

Trabalhos Futuros

Partindo deste trabalho, pode-se contribuir com o estado da arte de ferramentas de detecção de vulnerabilidades com:

- Desenvolvimento de uma base de SCs maior, rotulada de forma automática, a ser utilizada em conjunto com a base deste projeto
- Realização de um *survey* de ferramentas de detecção de vulnerabilidades, utilizando-se o *playground* completo de forma a levantar gaps no atual estado da arte
- Desenvolvimento de uma nova ferramenta de detecção de vulnerabilidades ou uma contribuição a uma ferramenta já presente na literatura de forma a fechar o gap identificado

Bibliografia

[1] Josselin Feist, Gustavo Grieco, and Alex Groce. "Slither: A Static Analysis Framework for Smart Contracts". doi: 10.1109/WETSEB.2019.00008.

[2] Chavhan Sujeet Yashavant, Saurabh Kumar, and Amey Karkare. ScrawlD: A Dataset of Real World Ethereum Smart Contracts Labelled with Vulnerabilities. 2022. arXiv: 2202.11409 [cs.CR].

[3] Zibin Zheng et al. DAppSCAN: Building Large-Scale Datasets for Smart Contract Weaknesses in DApp Projects. 2023. arXiv: 2305.08456 [cs.SE].

[4] Thomas Durieux et al. "Empirical Review of Automated Analysis Tools on 47,587 Ethereum Smart Contracts". doi: 10.1145/3377811.3380364

Integrantes: - Ryan Weege Achjian

Professor(a) Orientador(a): - Prof. Dr. Marcos Antônio Simplício Júnior

Co-orientador(a): -