Blockchain e Smart Contracts: vulnerabilidades que ainda acontecem no protocolo Proof of Stake

Alison Winkert Dos Santos
Centro Universitário Dinâmica das Cataratas
Foz do Iguaçu, Brasil
alisonwinkertds@gmail.com

Luciano Santos Cardoso
Centro Universitário Dinâmica das Cataratas
Foz do Iguaçu, Brasil
0009-0007-2998-8066

Alessandra Bussador
Centro Universitário Dinâmica das Cataratas
Foz do Iguaçu, Brasil
0000-0002-5900-9398

***Abstract*— Blockchain technology has evolved significantly since its inception, becoming a central pillar in the development of cryptocurrencies and decentralized systems. Initially, it provided a secure and immutable structure for digital transactions, leading to the creation and implementation of smart contracts that automate agreements and operations without the need for intermediaries. Over time, new methods have emerged to ensure the security and efficiency of these transactions, such as Ethereum's transition in 2022 from the Proof of Work (PoW) mechanism to Proof of Stake (PoS).**

***Keywords*—Blockchain; Smart Contracts; Vulnerability.**

***Resumo ou Resumen*— A tecnologia *Blockchain* tem evoluído significativamente desde sua concepção, tornando-se um pilar central no desenvolvimento de criptomoedas e sistemas descentralizados. Inicialmente, ela forneceu uma estrutura segura e imutável para transações digitais, o que levou à criação e implementação dos contratos inteligentes, que automatizam acordos e operações sem a necessidade de intermediários. Com o tempo, surgiram novas formas de garantir a segurança e eficiência dessas transações, como a transição do *Ethereum*, em 2022, do mecanismo *Proof of Work* (PoW) para *Proof of Stake* (PoS).**

***Palavras-chave*—Blockchain; Contratos Inteligentes; Vulnerabilidades.**

I. Introdução

Em 2008, Satoshi Nakamoto, cuja identidade real é desconhecida, lançou o conceito de *Blockchain* no artigo "*Bitcoin: A Peer-to-Peer Electronic Cash System*". A partir desse ponto, a popularidade das moedas descentralizadas cresceu significativamente [1].

Com a evolução do *Blockchain*, o desenvolvimento de *smart contracts* se tornou necessário devido ao aumento da procura e demanda por moedas descentralizadas. No entanto, o termo não foi publicado na mesma época; foi apresentado doze anos antes pelo criptografo Nick Szabo. Ele se baseou nos contratos comuns, nos quais estão envolvidas duas ou mais partes com o mesmo interesse e que necessitam formalizar aquele acordo o mais breve possível, que poderia ser modernizado por meios tecnológicos evitando futuras fraudes ou violações de contrato por parte do infrator. Contudo, ao longo deste amadurecimento em transações eletrônicas por contratos inteligentes, surgiu um conjunto de casos que revelavam vulnerabilidades nos algoritmos dos contratos, cujo objetivo era obter criptomoedas ou fraudar informações [2].

Este estudo tem como objetivo identificar vulnerabilidades no protocolo *Proof of Stake* aplicadas a *Smart Contracts*, destacando os riscos potenciais que os usuários dessa tecnologia podem enfrentar. Ao investigar o mecanismo *Proof of Stake*, o trabalho pretende testar falhas que possam comprometer a segurança dos *Smart Contracts*. A escolha da *blockchain* como foco é justificada por sua relevância e impacto na área de tecnologia, sendo um tema essencial para o desenvolvimento e a segurança de sistemas descentralizados.

II. Referencial Teórico

A. *Blockchain*

A tecnologia *Blockchain* emergiu como uma solução crucial no desenvolvimento de criptomoedas, oferecendo uma estrutura descentralizada que garante a segurança e a legitimidade dos ativos digitais. Assim como em qualquer sistema monetário, houve a necessidade de implementar mecanismos de segurança que evitassem a adulteração dos ativos, assegurando a autenticidade dos possuidores dos bens [3].

A estrutura da Blockchain é formada por um conjunto de blocos interligados por indicadores *hash*, que proporcionam segurança criptográfica. Esse modelo de registro é descentralizado e compartilhado entre diferentes nós da rede, permitindo que cada nó possua uma cópia do registro completo. A interligação dos blocos, feita através de criptografia, oferece uma barreira robusta contra possíveis ataques e fraudes [4][5].

A tecnologia *Blockchain* se torna necessária em situações que exigem a confiabilidade dos dados e recursos, especialmente quando várias partes interessadas estão envolvidas. Utilizando caminhos de auditoria confiáveis, é possível rastrear totalmente as transações registradas em um livro-razão digital, aumentando a transparência e aperfeiçoando o processo de prestação de contas [5].

B. *Smart Contract*

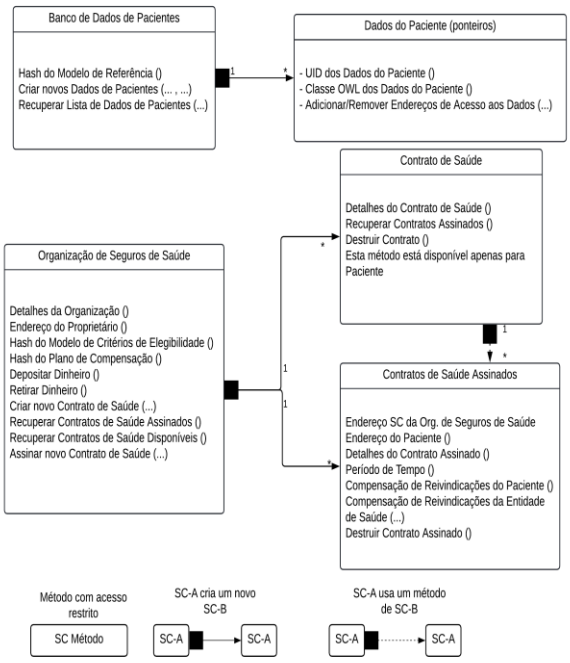
A seção abordada apresenta os detalhes dos contratos inteligentes desenvolvidos, suas interações internas e os serviços externos implementados para lidar com dados segurados. Cunhado originalmente como "contrato inteligente," este termo refere-se a um programa de computador que executa automaticamente os termos de um contrato. O conceito foi introduzido por Nick Szabo em 1996, no artigo "*SmartContracts: Building Blocks for Digital Free Markets*". Szabo pro-

pôs que uma variedade de cláusulas contratuais, como garantias e direitos de propriedade, poderia ser integrada ao hardware e software, tornando o descumprimento do contrato custoso ou até mesmo proibitivo [6].

Os contratos inteligentes são uma parte crucial do *blockchain*, pois este não só fornece um registro distribuído e imutável de todos os eventos, mas também permite a criação de código de computador que define objetivamente como um processo será gerenciado e quais etapas serão tomadas em resposta a eventos específicos. Um dos principais objetivos dos contratos inteligentes propostos no *Ethereum* foi superar as limitações do *Bitcoin*[8][7].

A Figura 1 oferece uma visão geral dos cinco contratos inteligentes desenvolvidos, detalhando os métodos disponíveis para cada um e seu escopo (por exemplo, público ou privado). As interações entre os contratos são indicadas por setas, e o armazenamento de criptomoedas é representado por um símbolo no canto superior direito dos contratos que possuem essa funcionalidade. As criptomoedas armazenadas são controladas inteiramente pela plataforma *blockchain* (no caso, *Ethereum*), e os usuários podem depositar ou retirar fundos através dos métodos apropriados disponíveis somente para os usuários autorizados [9].

Fig. 1. *Smart Contracts* adaptado de Chondrogiannis *et al*.



C. *Proof of Stake*

Em 15 de setembro de 2022, o Ethereum passou por uma importante transição de seu mecanismo de consenso de *Proof of Work* (PoW) para *Proof of Stake* (PoS), como uma forma de resistência *Sybil*, conforme discutido por [10]. Com essa mudança, a arquitetura do Ethereum passou a operar com duas camadas distintas: a camada de execução e a camada de consenso [11].

A camada de execução, que guarda semelhanças com o antigo protocolo PoW, continua a ser responsável pela validação e execução das transações na rede. Em contraste, a camada de consenso, que foi construída sobre a *beacon chain*, tem como foco principal alcançar consenso entre os validadores, papel que anteriormente era desempenhado pelos mineradores tradicionais no sistema PoW [11][12].

No paradigma PoS, os validadores assumem o lugar dos mineradores tradicionais. Este novo mecanismo de consenso foi desenvolvido com o propósito de melhorar o algoritmo PoW, que era considerado ineficiente e vulnerável a comportamentos inadequados por parte dos delegados, segundo a análise dos autores [12][13].

D. *Segurança*

A blockchain surgiu como uma inovação em sistemas de consenso distribuído, permitindo o armazenamento seguro e a verificação de transações e outros dados sem a necessidade de uma autoridade central. Originalmente, o conceito de blockchain estava intimamente ligado ao mecanismo de prova de trabalho (*Proof of Work* - PoW) baseado em *hash*, utilizado pelo Bitcoin, que é amplamente reconhecido atualmente. Com o tempo, surgiram mais de uma centena de blockchains alternativas. Algumas são variações simples do modelo do *Bitcoin*, enquanto outras diferem significativamente em seu design, oferecendo diferentes níveis de funcionalidade e segurança. Isso reflete os esforços da comunidade de pesquisa em busca de uma tecnologia blockchain que seja mais simples, escalável e prática para implementação [14].

Os contratos inteligentes, uma componente essencial das blockchains, são distribuídos para todos os mineradores na rede e, uma vez ativados, são executados autonomamente por cada nó, sem necessidade de interação do usuário, facilitando assim uma troca justa. Esses contratos são armazenados na blockchain como *bytecode*, o que os torna ilegíveis para humanos. Embora existam alguns códigos-fonte disponíveis em sites como *Etherchain* e *Etherscan*, o envio desses códigos não é obrigatório, resultando em bibliotecas incompletas [15].

Para mitigar ataques distribuídos de negação de serviço (DDoS), como programas infinitos, as plataformas de contratos inteligentes implementam um mecanismo de "gás", que cobra dos remetentes por uso de computação e armazenamento. Embora esse seja um método eficaz para proteger o sistema, contratos inteligentes podem ser configurados para consumir mais gás do que o necessário, prejudicando os usuários [16].

Os contratos *honeypot* são um tipo específico de contrato inteligente na plataforma *Ethereum*, criados intencionalmente pelos desenvolvedores com falhas embutidas para atrair atacantes que buscam explorar vulnerabilidades. Diferente dos contratos inteligentes convencionais, que são projetados com o foco em segurança, os contratos *honeypot* visam enganar os atacantes. Para explorar as vulnerabilidades percebidas nesses contratos, os atacantes geralmente precisam transferir uma quantia específica de *ether*. No entanto, após

realizar a transferência, o contrato não opera como o atacante esperava, e o valor investido não pode ser recuperado [17].

O sucesso dos contratos *honeypot* se deve ao fato de que muitos atacantes, movidos pela ganância, não consideram que podem estar caindo em uma armadilha, preparada por desenvolvedores mais experientes. A fim de aumentar a eficácia do *honeypot*, os criadores frequentemente tornam o código-fonte desses contratos público, facilitando a detecção e tornando-os "vulneráveis". Isso faz com que os atacantes sejam induzidos a agir sem a devida cautela, acreditando que estão explorando uma fraqueza genuína no código [17].

Este método de exploração dos contratos inteligentes exemplifica como falhas deliberadamente introduzidas podem ser usadas de forma estratégica para enganar aqueles que tentam burlar o sistema, revelando um novo nível de complexidade e armadilhas na dinâmica de segurança em plataformas *blockchain* como o *Ethereum* [18].

A Figura 2 demonstra as diferentes partes e etapas envolvidas em um *honeypot*. Esse tipo de contrato normalmente opera em três fases principais:

O atacante cria um contrato que aparenta ter uma vulnerabilidade, utilizando uma quantia como isca para atrair vítimas;

A vítima, ao tentar explorar a suposta falha no contrato, transfere pelo menos o valor mínimo exigido, mas não consegue obter sucesso na exploração;

O atacante então desfaz a armadilha, recuperando os recursos que a vítima desperdiçou em sua tentativa de exploração.

Importante destacar que não é necessário que o atacante possua habilidades técnicas avançadas para criar um *honeypot*. Na verdade, o atacante tem capacidades semelhantes às de um usuário comum do Ethereum. Ele ou ela apenas precisa dos recursos adequados para implementar o contrato inteligente e configurar a armadilha.

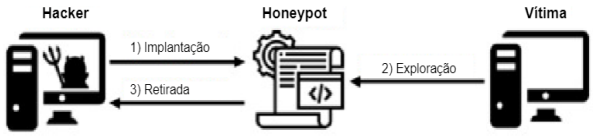


Fig. 2. Fases do contrato *Honeypot* de Torres *et al*.

O contrato ilustrado na Figura 3 exemplifica um tipo de "*honeypot*" que utiliza a técnica chamada de "desordem de saldo". Nesse contrato, a função "*multiplicate*" parece sugerir que o saldo do contrato (*this.balance*) e o valor enviado na transação (*msg.value*) serão transferidos para um endereço aleatório, desde que o valor enviado pelo usuário seja igual ou superior ao saldo atual do contrato inteligente. Essa lógica pode levar um usuário inexperiente a acreditar que, ao enviar uma quantia maior ou igual ao saldo do contrato, ele poderá recuperar o valor "investido" juntamente com o saldo do contrato [18].

No entanto, ao tentar essa estratégia, o usuário descobrirá rapidamente que a linha 5 do código não é executada porque a condição na linha 4 não é satisfeita. Isso ocorre porque o saldo do contrato já foi incrementado pelo valor da transação antes da execução completa do contrato inteligente. Existem alguns pontos importantes a considerar:

1) a condição na linha 4 pode ser atendida se o saldo atual do contrato for zero, mas nesse cenário, o usuário não teria motivos para interagir com o contrato;

2) a soma *this.balance* + *msg.value* na linha 5 apenas reforça a falsa percepção do usuário de que o saldo do contrato é atualizado somente após a execução, quando na verdade, ele é alterado imediatamente após a transação.

```
1  contract MultiplicatorX3 {
2
3      function multiplicate ( address adr) payable {
4          if ( msg . value ≥ this . balance )
5          adr . transfer ( this . balance + msg . value );
6      }
7  }
```

Fig. 3. Fases do contrato *Honeypot* de Rodrigues *et al*.

III. Metodologia

Os procedimentos metodológicos descritos para a pesquisa centram-se na análise e exploração de vulnerabilidades em contratos inteligentes na blockchain Ethereum, utilizando uma combinação de revisão bibliográfica, análise de código e testes práticos. O estudo busca identificar falhas, explorar suas consequências e propor soluções para mitigar os riscos associados. A metodologia segue um planejamento rigoroso, incluindo a escolha do tema, coleta de dados, formulação de hipóteses e seleção de métodos e técnicas apropriados [2].

A pesquisa é classificada como aplicada, com enfoque descritivo, visando testar problemas práticos e melhorar processos existentes, particularmente no que se refere à segurança de contratos inteligentes [19]. A abordagem adotada é mista, combinando métodos qualitativos para interpretar as vulnerabilidades e quantitativos para medir a frequência e impacto dessas falhas [2]. O cenário da pesquisa é a plataforma *Ethereum*, com foco em contratos escritos em *Solidity*.

Os contratos inteligentes analisados foram selecionados com base em sua relevância na comunidade Ethereum e histórico de vulnerabilidades conhecidas. A coleta de dados envolve o uso de ferramentas específicas para o desenvolvimento e teste de contratos inteligentes, além de uma revisão detalhada da literatura existente. A análise qualitativa e quantitativa dos dados permitirá uma compreensão aprofundada das vulnerabilidades e das práticas recomendadas para melhorar a segurança e confiabilidade dos contratos inteligentes [19].

IV. Conclusão

Ao decorrer do trabalho, existiram circunstâncias em volta do quais meios seriam utilizados para o desenvolvimento do protótipo que faça a simulação de uma eventual invasão nas falhas do protocolo *Proof of Stake*. Uma das ideias seria testar vulnerabilidades que aconteceram no PoW usando *blockchain* que armazenam contratos inteligentes com *Ethereum*, assim, fazendo o uso de meios como por exemplo, como ataque de reentrância utilizado no caso *The DAO Hack* para analisar se o protocolo conseguiu sanar o problema ocorrido anteriormente.

Os testes ainda não foram realizados porque o protótipo está em fase de desenvolvimento a partir da conclusão desta fase. Podendo ainda sofrer alterações ou implementado novas informações e ações no protótipo.

Com trabalhos futuros, serão apresentados o resultado destas comparações e uma metodologia de testes de vulnerabilidade existente na metodologia PoS.

Referências

[1] NAKAMOTO, Satoshi *et al*. Bitcoin: A Peer-to-Peer Electronic Cash System. [*s. l.*], p. 1-9, 2008. Disponível em: https://bitcoin.org/bitcoin.pdf. Acesso em: 29 out. 2024.

[2] RODRIGUES, Fillipe Barros *et al*. Análise e exploração de vulnerabilidades em smart contracts baseados em blockchain Ethereum. *In*: RODRIGUES, Fillipe Barros *et al*. **Análise e exploração de vulnerabilidades em smart contracts baseados em blockchain Ethereum**. 2019. TRABALHO DE GRADUAÇÃO (Engenheiro de Redes de Comunicação) - UNIVERSIDADE DE BRASILIA, Faculdade de Tecnologia, Brasília, 2019. p. 84. Disponível em: https://bdm.unb.br/bitstream/10483/29193/1/2019_FillipeRodrigues_LucasJoseAmaro_tcc.pdf. Acesso em: 29 out. 2024.

[3] RIBEIRO, Lucas *et al*. Introdução à Blockchain e Contratos Inteligentes, [*s. l.*], p. 1-56, 2021. Disponível em: https://repositorio.ufsc.br/handle/123456789/221495. Acesso em: 29 out. 2024.

[4] ALLADI, Tejasvi *et al*. Blockchain in Smart Grids: A Review on Different Use Cases. **Surveys of Sensor Networks and Sensor Systems Deployments**, [*s. l.*], v. 19, p. 1-25, 8 nov. 2019. DOI https://doi.org/10.3390/s19224862. Disponível em: https://www.mdpi.com/1424-8220/19/22/4862. Acesso em: 29 out. 2024.

[5] JOSÉ MUCIO MONTEIRO, M. et al. TRIBUNAL DE CONTAS DA UNIÃO REPÚBLICA FEDERATIVA DO BRASIL. [s.l: s.n.]. Disponível em: <www.tcu.gov.br>.

[6] SZABO, Nick. Smart Contracts: Building Blocks for Digital Market. , [s. l.], p. 1-11, 1996. Disponível em: https://www.truevaluemetrics.org/DBpdfs/BlockChain/Nick-Szabo-Smart-Contracts-Building-Blocks-for-Digital-Markets-1996-14591.pdf. Acesso em: 29 out. 2024.

[7] GUO, Huaqun *et al*. A survey on blockchain technology and its security. **Blockchain: Research and Applications**, [*s. l.*], v. 3, p. 2901-2925, 12 fev. 2022. DOI https://doi.org/10.1016/j.bcra.2022.100067. Disponível em: https://www.sciencedirect.com/science/article/pii/S2096720922000070. Acesso em: 29 out. 2024.

[8] KHAN, Shafaq Naheed *et al*. Blockchain smart contracts: Applications, challenges, and future trends. **Peer-to-Peer Networking and Applications**, Malaysia, v. 14, p. 2901-2925, 18 abr. 2021. DOI 10.1007/s12083-021-01127-0. Disponível em: https://link.springer.com/article/10.1007/s12083-021-01127-0. Acesso em: 29 out. 2024.

[9] CHONDROGIANNIS, Efthymios; ANDRONIKOU, Vassiliki; KARANASTASIS, Efstathios; LITKE, Antonis; VARVARIGOU, Theodora. Using blockchain and semantic web technologies for the implementation of smart contracts between individuals and health insurance organizations. **Blockchain: Research and Applications**, Malaysia, v. 3, p. 1-14, 2022. DOI https://doi.org/10.1016/j.bcra.2021.100049. Disponível em: https://www.sciencedirect.com/science/article/pii/S2096720921000440. Acesso em: 29 out. 2024.

[10] ETHEREUM. The Merge. Disponível em: <https://ethereum.org/en/roadmap/merge/>. Acesso em: 4 jul. 2024.

[11] GRANDJEAN, Dominic *et al*. Ethereum Proof-of-Stake Consensus Layer: Participation and Decentralization. **Distributed, Parallel, and Cluster Computing**, Malaysia, ano 2020, v. 10, p. 1-26, 22 nov. 2023. DOI https://doi.org/10.48550/arXiv.2306.10777. Disponível em: https://arxiv.org/abs/2306.10777. Acesso em: 29 out. 2024.

[12] SAAD, Sheikh Munir Skh *et al*. Comparative Review of the Blockchain Consensus Algorithm Between Proof of Stake (POS) and Delegated Proof of Stake (DPOS). **International Journal of Innovative Computing**, Malaysia, ano 2020, v. 10, p. 27-32, 19 nov. 2020. DOI https://doi.org/10.11113/ijic.v10n2.272. Disponível em: https://ijic.utm.my/index.php/ijic/article/view/272. Acesso em: 29 out. 2024.

[13] BAMAKAN, Seyed Mojtaba Hosseini *et al*. A survey of blockchain consensus algorithms performance evaluation criteria. **Expert Systems with Applications**, Zurich, ano 2020, v. 154, p. 1-18, 21 abr. 2020. DOI https://doi.org/10.1016/j.eswa.2020.113385. Disponível em: https://www.sciencedirect.com/science/article/abs/pii/S0957417420302098. Acesso em: 29 out. 2024.

[14] GHASSAN, Karame *et al*. Blockchain Security and Privacy. **Blockchain Security and Privacy**, Zurich, ano 2018, v. 16, p. 11-12, 10 abr. 2018. DOI 10.1109/MSP.2018.3111241. Disponível em: https://www.computer.org/csdl/magazine/sp/2018/04/msp2018040011/13rRUxBJhE9. Acesso em: 29 out. 2024.

[15] LIU, Jing *et al*. A Survey on Security Verification of Blockchain Smart Contracts. **A Survey on Security Verification of Blockchain Smart Contracts**, Oxford, UK, ano 2019, v. 7, p. 77894-77904, 10 jun. 2019. DOI 10.1109/ACCESS.2019.2921624. Disponível em: https://ieeexplore.ieee.org/document/8732934. Acesso em: 29 out. 2024.

[16] WANG, Zeli *et al*. Ethereum smart contract security research: survey and future research opportunities. **Ethereum smart contract security research: survey and future research opportunities.**, Oxford, UK, ano 2021, v. 15, p. 1-18, 15 abr. 2021. DOI 10.1007/s11704-020-9284-9. Disponível em: https://journal.hep.com.cn/fcs/EN/10.1007/s11704-020-9284-9#2. Acesso em: 29 out. 2024.

[17] CHEN, Weili *et al*. **Honeypot Contract Risk Warning on Ethereum Smart Contracts**, Oxford, UK, ano 2020, p. 1-8, 3 jun. 2020. DOI 10.1109/JCC49151.2020.00009. Disponível em: https://ieeexplore.ieee.org/document/9183392. Acesso em: 29 out. 2024.

[18] TORRES, Christof Ferreira *et al*. **The Art of The Scam: Demystifying Honeypots in Ethereum Smart Contracts**, Santa Clara, CA, ano 2019, p. 1591-1607, 14 ago. 2019. Disponível em: https://www.usenix.org/system/files/sec19-torres.pdf. Acesso em: 29 out. 2024.

[19] FERREIRA, Frederico Lage *et al*. **Blockchain e Ethereum Aplicações e Vulnerabilidades**, São Paulo, ano 2017, p. 1-36, 27 nov. 2017. Disponível em: https://linux.ime.usp.br/~fredlage/mac0499/Monografia.pdf. Acesso em: 29 out. 2024.