Pagamentos Automatizados Programáveis em Carteiras Auto-Custodiadas: Uma Exploração Técnica

Ana Julia Bittencourt Fogaça[1], **Saulo Popov Zambiasi**[2]

[1]Universidade do Sul de Santa Catarina (UNISUL)
Tubarão - SC - Brasil

[2]Universidade do Sul de Santa Catarina (UNISUL)
Florianópolis - SC - Brasil

anajuliabit@gmail.com, saulopz@gmail.com

Abstract.

1. Introdução

A tecnologia blockchain, primeiramente introduzida por Satoshi Nakamoto em 2008, é identificada como uma megatendência computacional capaz de revolucionar múltiplos setores industriais[World Economic Forum]. As características distintas de segurança, transparência e rastreabilidade inerentes à blockchain têm incentivado uma ampla gama de setores a explorar seu uso na reestruturação de suas operações fundamentais. A aplicabilidade dessa tecnologia ultrapassa o domínio das criptomoedas, abarcando setores como pagamentos, gerenciamento de identidade, saúde, eleições governamentais e outros[Pu and Lam 2021].

A publicação do whitepaper do Ethereum em 2014 marcou um progresso notável no desenvolvimento da tecnologia blockchain[Buterin 2014]. Ao contrário do Bitcoin, que foi inicialmente projetado como um sistema de pagamento digital, o Ethereum introduziu uma funcionalidade revolucionária na tecnologia blockchain: os contratos inteligentes. A principal inovação proporcionada pelo Ethereum é a integração de uma máquina virtual capaz de executar códigos em linguagens de programação *Turing complete* na blockchain, possibilitando a criação de aplicativos descentralizados. Esses aplicativos substituem o sistema de back-end por contratos inteligentes que operam em uma blockchain[Dannen 2017]. No entanto, apesar de seu potencial imenso, a complexidade associada à aplicação prática desta tecnologia é um dos obstáculos para sua adoção em larga escala[Sadhya and Sadhya 2018]. Para utilizar os aplicativos descentralizados (DApps) de maneira segura, os usuários precisam ter conhecimento técnico em criptografia para manter suas chaves privadas seguras[Antonopoulos and Wood 2018], devido à natureza das carteiras auto-custodiadas na blockchain, entraremos em mais detalhes a seguir. Esta não é a realidade para a maioria dos usuários da internet, o que dificulta a adoção de DApps. Além disso, a experiência do usuário final é insatisfatória quando comparada à maneira como usamos a internet hoje. É como se fosse necessário inserir sua senha (chave privada) para cada ação que você realiza que não seja apenas consumir dados - em termos técnicos, uma operação de leitura.

Em resposta ao desafio emergente, propostas inovadoras são regularmente introduzidas com a intenção de aprimorar a experiência do usuário e, por extensão, promover a adoção de DApps. Uma dessas propostas recentes é a *Account Abstrac-*

tion (AA). AA apresenta uma inovação significativa, proporcionando maior flexibilidade no processo de validação de transações e permitindo autorizações personalizadas que não dependem necessariamente de uma assinatura criptográfica. Esta funcionalidade tem o potencial de aprimorar consideravelmente a experiência do usuário no ecossistema Ethereum[Coyle 2023], abrindo caminho para novos casos de uso que anteriormente eram inviáveis dentro de uma blockchain. Um desses casos de uso foi recentemente o foco de investigação da Visa, uma empresa líder em soluções de pagamento. A Visa sugeriu a possibilidade de realizar pagamentos automáticos na blockchain sem a necessidade de exposição de chaves privadas, aproveitando os benefícios da AA[Andrew Beams and Kumaresan 2023]. Contudo, os detalhes técnicos ou o código-fonte para tal implementação não foram divulgados. Esta pesquisa se propõe a explorar profundamente essa funcionalidade e propor uma implementação prática para viabilizar pagamentos automáticos em carteiras auto-custodiadas. O objetivo é fornecer uma perspectiva esclarecedora e diretrizes pragmáticas que possam servir como referência para futuras aplicações de pagamentos automatizados em blockchains como o Ethereum.

A estrutura deste artigo é organizada da seguinte maneira: Seguindo esta introdução, avançamos para uma revisão bibliográfica abrangente, iniciando com uma explanação aprofundada do funcionamento do Ethereum, esboçando as responsabilidades da Ethereum Virtual Machine (EVM), elucidando os diferentes tipos de contas e esclarecendo as propostas da AA. Adicionalmente, empreendemos uma análise do estudo conduzido pela Visa. Na seção de desenvolvimento, propomos uma implementação em Solidity com o intuito de habilitar pagamentos automáticos programáveis em carteiras auto-custodiadas. A nossa discussão culmina com uma reflexão sobre as implicações e possíveis aplicações desta nova funcionalidade nas blockchains.

2. Revisão Bibliográfica

Como mencionado anteriormente, a característica distintiva do Ethereum reside em sua capacidade de executar códigos, mais especificamente, contratos inteligentes, em uma máquina virtual denominada *Ethereum Virtual Machine* (EVM). A EVM, uma máquina virtual global que opera em um formato de instância única, e é executada repetidamente em uma variedade de computadores em todo o mundo. Cada nó no Ethereum mantem uma cópia local da EVM, que é responsável por validar e executar contratos inteligentes. Qualquer mudança de estado decorrente da execução desses contratos inteligentes é registrada no Ethereum[Antonopoulos and Wood 2018].O Ethereum tem como objetivo permitir a execução de aplicações e scripts arbitrários que operam em transações, utilizando uma blockchain para sincronizar o estado global de maneira totalmente verificável por qualquer participante do sistema[Hildenbrandt et al. 2018]. O estado, que é único e compartilhado entre todos os nós, confere uma característica essencial aos contratos inteligentes: a necessidade de serem determinísticos. A linguagem mais popular para a criação de contratos inteligentes é o Solidity[Solidity contributors]. O Solidity é uma linguagem de programação de alto nivel e precisa ser compilada para código EVM - bytecode que a EVM pode executar nativamente[Wood et al. 2014]. Devido a essas características, o Ethereum é frequentemente descrito como um 'computador mundial de propósito geral'[Antonopoulos and Wood 2018].

A compreensão clara das diferenças entre os dois tipos de contas no Ethereum é fundamental para um entendimento completo do funcionamento do Ethereum. No Ethe-

reum, existem duas categorias distintas de contas: as contas de propriedade externa (*externally owned accounts* - EOAs), que são controladas por chaves privadas, e as contas de contrato (*contract accounts* - CAs), que são controladas pela lógica incorporada no código dos contratos inteligentes associados a elas[Buterin 2014]. Em termos técnicos, as CAs têm a propriedade 'codeHash' - o hash do código do contrato inteligente na linguagem da EVM, que é o código executado sempre que uma CA é o destinatário de uma transação, e o campo 'storageRoot' - que representa o estado do contrato inteligente preenchidos. Em contraste, as EOAs têm ambos os campos vazios[Wood et al. 2014].

O fato de as Externally Owned Accounts (EOAs) serem controladas unicamente por uma chave privada que emprega o esquema de assinatura ECDSA[National Institute of Standards and Technology 2013] para autenticar transações restringe a utilidade da blockchain. Isso exige que os usuários possuam conhecimento técnico em armazenamento de chaves privadas e na tarefa de assinar criptograficamente cada ação que desejam que resulte na alteração do estado global da blockchain. Esta tarefa não é trivial para muitos, evidenciado pela popularidade de carteiras com maior grau de segurança, como as carteiras multisig e as carteiras físicas como a Ledger Nano S[**?**] e a Trezor[**?**] para gerenciar contas de blockchains. Outro ponto crucial a destacar é que o algoritmo ECDSA pode se tornar obsoleto com a computação quântica[Oder et al. 2014], o que sublinha a urgência de buscar alternativas.

Como o Ethereum é uma rede descentralizada e de código aberto, a comunidade constantemente formula propostas de melhoria conhecidas como Ethereum Improvement Proposals (EIPs). A EIP-4337[Buterin et al. 2021], também denominada Account Abstraction (AA), introduz uma inovação radical no modelo convencional de contas no Ethereum, sugerindo a utilização de contratos inteligentes em vez das tradicionais EOAs. Esta proposta tem o potencial de abrir novos casos de uso na plataforma.

Ao proporcionar maior flexibilidade para as contas de usuários, AA desencadeia uma série de novas capacidades, destacando-se a possibilidade de autorização personalizada sem a obrigatoriedade do uso de ECDSA, assim como nas EOAs. Isso permite a adaptação da lógica de autorização de acordo com necessidades específicas. Através da EIP-4337, podemos estabelecer regras de validação que não dependem necessariamente da assinatura do proprietário da conta, o que representa uma inovação para as blockchains. Até então, a autorização de transações era exclusivamente baseada na atomicidade das assinaturas criptográficas, exigindo que a aprovação ocorresse instantaneamente, sem a possibilidade de pré-aprovação.Considerando que a EIP-4337 é uma proposta recente, concebida no final de 2021 e oficialmente implementada em março de 2023, há uma lacuna notável em estudos acadêmicos que aprofundem seu funcionamento. Portanto, é imperativo que nos aprofundemos nos detalhes técnicos da AA para compreender como ela pode ser empregada para viabilizar pagamentos automáticos programáveis em carteiras autogerenciadas.

Referências

Andrew Beams, Catherine Gu, S. R. M. M. and Kumaresan, R. (2023). Auto payments for self-custodial wallets.

Antonopoulos, A. and Wood, G. (2018). *Mastering Ethereum: Building Smart Contracts and DApps*. O'Reilly Media.

Buterin, V. (2014). Ethereum white paper.

Buterin, V., Weiss, Y., Gazso, K., Patel, N., Tirosh, D., Nacson, S., and Hess, T. (2021). Erc-4337: Account abstraction using alt mempool. Ethereum Improvement Proposals.

Coyle, K. (2023). Account abstraction: Use cases, technical overview, and security considerations.

Dannen, C. (2017). *The EVM*, pages 47–67. Apress, Berkeley, CA.

Hildenbrandt, E., Saxena, M., Rodrigues, N., Zhu, X., Daian, P., Guth, D., Moore, B., Park, D., Zhang, Y., Stefanescu, A., and Rosu, G. (2018). *KEVM: A Complete Formal Semantics of the Ethereum Virtual Machine*, pages 204–217.

National Institute of Standards and Technology (2013). Digital signature standard. Technical Report NIST.FIPS.186-4, National Institute of Standards and Technology.

Oder, T., Pöppelmann, T., and Güneysu, T. (2014). Beyond ecdsa and rsa: Lattice-based digital signatures on constrained devices. In *Proceedings of the 51st Annual Design Automation Conference*, DAC '14, page 1–6, New York, NY, USA. Association for Computing Machinery.

Pu, S. and Lam, J. S. L. (2021). Blockchain adoptions in the maritime industry: a conceptual framework. *Maritime Policy & Management*, 48(6):777–794.

Sadhya, V. and Sadhya, H. (2018). Barriers to adoption of blockchain technology.

Solidity contributors. Solidity documentation. `https://docs.soliditylang.org/en/v0.8.20/`.

Wood, G. et al. (2014). Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, 151(2014):1–32.

World Economic Forum. Deep shift - technology tipping points and societal impact. `http://www3.weforum.org/docs/WEF_GAC15_Technological_Tipping_Points_report_2015.pdf`.