Proceeding Series of the Brazilian Society of Computational and Applied Mathematics
Trabalho apresentado no XLIV CNMAC, Fundação Getulio Vargas - Rio de Janeiro - RJ, 2025.

Implementação de um Modelo Computacional de Troca Justa Utilizando Ambiente de Execução Confiável

Dhileane A. Rodrigues[1], Fabricia C. Roos-Frantz[2], Rafael Z. Frantz[3], Sandro Sawicki[4]
Unijuí, Ijui, RJ
Carlos Molina-Jimenez[5]
Cambridge, Cambridge, UK

Resumo. Este artigo propõe um protocolo de troca justa com TTP descentralizado, que pode fazer uso de TEEs como Intel SGX, TrustZone ARM e AMD Secure Memory Encryption, para garantir a integridade das transações. A solução oferece maior privacidade e autonomia aos participantes, com a divisão do protocolo em attestables e PBB, permitindo uma implementação flexível e distribuída. Em síntese, este artigo apresenta uma abordagem que introduz um protocolo de troca justa com TTP descentralizado utilizando TEEs, visando garantir a privacidade e a autonomia dos participantes de uma troca.

Palavras-chave. Troca Justa, Ambientes de Execução Confiáveis, Sistemas Distribuídos.

1 Introdução

Uma troca justa é um problema clássico de sistemas distribuídos, onde é necessário alcançar consenso entre as partes envolvidas para garantir um resultado satisfatório. Em transações eletrônicas, a troca justa visa assegurar que ambas as partes entreguem seus bens ou serviços de forma simultânea, eliminando o risco de prejuízo para uma das partes. Esse conceito é fundamental para estabelecer a confiança mútua, essencial para transações bem-sucedidas, como no exemplo clássico de Alice e Bob, em que ambos dependem de garantias para assegurar uma troca justa. A busca por soluções para esse problema gerou o desenvolvimento de protocolos de troca justa, sendo um tema de pesquisa contínua desde 1997 [3].

Esses protocolos garantem a segurança e a privacidade das transações, assegurando que ambas as partes recebam o que foi acordado, sem riscos de fraude. Eles podem ser divididos em dois componentes principais: o protocolo de troca, que define como os itens serão trocados, e o protocolo de sincronização, que verifica se a troca foi realizada corretamente. A sincronização é crucial para determinar a validade da troca, abordando questões como interceptação ou falha no processo. Esses aspectos são essenciais para garantir a integridade da transação e prevenir fraudes [9].

Historicamente, soluções baseadas em protocolos de *escrow-based fair exchange* têm sido amplamente empregadas para assegurar a justiça nas transações. Um TTP (*Trusted Third Party*) é designado para coletar e distribuir os itens, mas essa abordagem centralizada é vulnerável a ataques e falhas. A confiança em um único intermediário central levanta preocupações sobre a segurança e privacidade das partes envolvidas. Embora seja uma solução prática, o TTP centralizado apresenta um risco considerável, pois sua integridade é essencial para a justiça da troca. Isso gerou interesse por alternativas que busquem mitigar as falhas associadas a sistemas centralizados [8, 14].

Com os avanços tecnológicos, surgiu a ideia de descentralizar o TTP, utilizando ambientes de execução confiáveis (TEEs), que oferecem maior segurança. Os TEEs protegem o processo de troca contra invasões e garantem a integridade e privacidade dos itens transacionados. Eles permitem a execução de operações de forma isolada e protegida, sem depender de um único TTP. Essa abordagem descentralizada representa uma melhoria significativa em termos de segurança e eficiência, eliminando o risco de falhas decorrentes da dependência de um único intermediário. A proposta deste artigo apresenta um protocolo de troca justa utilizando TEEs, garantindo que a troca seja realizada de forma justa e segura, sem recorrer a um TTP tradicional [11].

[1]dhileane.rodrigues@sou.unijui.edu.br
[2]frfrantz@unijui.edu.br
[3]rzfrantz@unijui.edu.br
[4]sawicki@unijui.edu.br
[5]cm770@cam.ac.uk

2 Trabalhos Relacionados

O problema da troca justa está relacionado à falta de confiança mútua entre as partes envolvidas. Este estudo revisa uma ampla gama de pesquisas e desenvolvimentos de protocolos de troca justa, abrangendo o período de 1997 a 2024. A seguir, são descritos alguns dos principais estudos que fundamentaram a evolução da nossa pesquisa.

Em 1997, Asokan, Schunter e Waidner propuseram um protocolo de troca segura baseado em criptografia assimétrica e assinaturas digitais arbitrárias, utilizando um TTP centralizado para coordenar a troca [3]. No ano seguinte, Asokan, Shoup e Waidner apresentaram protocolos otimistas de troca justa que minimizam a participação do TTP, ativando-o apenas em casos de erros ou disputas. Esses protocolos assumem a honestidade das partes, com mecanismos de segurança para proteger a transação contra ações maliciosas [4].

Outros estudos contribuíram com avanços importantes, como o trabalho de Pagnia et al. (2000), que desenvolveu um protocolo de troca justa utilizando *Trusted Processing Environments* (TPE) e agentes de troca justa (FEA), assegurando a integridade e segurança nas transações online [15]. Já Avoine et al. (2003) propuseram um protocolo de commit atômico em módulos de segurança confiáveis, resolvendo problemas de troca justa no nível de hosts não confiáveis [5].

Pesquisas mais recentes exploraram o uso de novas tecnologias, como o trabalho de Ersoy et al. (2021), que introduziu um protocolo de troca de bens digitais em Ethereum, com custos de execução competitivos [7]. Além disso, Kuccuk et al. (2016) investigaram o uso de Intel SGX para criar uma Trusted Remote Entity (TRE), permitindo que múltiplos participantes computem dados privados sem revelá-los [13]. Os trabalhos de Abadi et al. (2023) e Zhang et al. (2024) também contribuíram para o avanço na construção de protocolos seguros de troca justa baseados em blockchain e criptografia por atributos [1, 18].

3 Protocolo com TTPs Centralizados

Em protocolos com TTPs centralizados, um único TTP media a troca de informações ou recursos entre duas partes, garantindo que a transação seja realizada de forma justa e segura. Nos protocolos de troca justa baseados em custódia (escrow), o TTP executa todas as operações essenciais, funcionando como mediador entre Alice e Bob, garantindo que ambos compartilhem a mesma visão sobre o status da troca (se foi liberada ou restaurada). A sincronização é controlada pelo TTP, que desempenha um papel central em evitar disputas. A principal desvantagem desses sistemas é a vulnerabilidade a ataques de negação de serviço (DoS), devido à constante necessidade de envolvimento do TTP nas transações.

Em abordagens centralizadas, como a utilizada pela Amazon, o TTP é responsável por autenticar partes, armazenar dados e intermediar transações, mas gera preocupações com a privacidade dos dados e custos elevados. Por outro lado, os protocolos de troca justa otimista, que pressupõem a boa fé das partes envolvidas, permitem que Alice libere o item diretamente para Bob, sem verificações rigorosas. No entanto, para resolver disputas, o TTP é utilizado. Nesses protocolos, distinguem-se a troca justa fraca no *front-end*, onde Alice inicia a troca, e a troca justa forte no *back-end*, onde o TTP garante que a transação seja validada ou cancelada, dependendo do comportamento das partes.

Atualmente, muitas soluções no mercado usam TTPs centralizados para facilitar transações, como a Amazon no comércio eletrônico. A Amazon age como um TTP central, autenticando as partes, armazenando dados confidenciais e intermediando as transações. Além disso, valida transações, resolve disputas e mantém a conformidade e auditoria para garantir a confiabilidade. Em nossa abordagem, a Amazon desempenha o papel de TTP, verificando e validando informações, enquanto Alice e Bob representam o comprador e o vendedor, respectivamente. No entanto, tais abordagens centralizadas geram preocupações com a privacidade de dados e aumento de custos.

Os problemas de privacidade com TTPs centralizados incluem a coleta e uso de dados pessoais dos clientes, compartilhamento desses dados com terceiros e monitoramento das atividades de compra dos usuários. Isso destaca a necessidade de desenvolver mecanismos para proteger a privacidade e os dados pessoais dos usuários nas transações.

4 Protocolos de Troca Justa de Liberação Gradual

Em situações onde Alice e Bob não têm um terceiro confiável para mediar sua troca, surge a necessidade de protocolos que garantam equidade, sendo os protocolos de liberação gradual uma solução para isso [16]. Nessa abordagem, a troca de itens é realizada de forma progressiva, trocando pequenas partes

dos itens, como pixels de uma imagem, letras de um texto ou bytes de um arquivo, ao invés de fazer a troca completa de uma vez. A principal característica desses protocolos é a operação de sincronização, que assegura que ambas as partes tenham uma visão compartilhada e consistente do processo de troca, coordenando a liberação de informações ou privilégios ao longo do tempo. Isso evita discrepâncias que poderiam resultar em uma troca injusta. No entanto, a principal desvantagem dos protocolos de liberação gradual é que exigem que ambas as partes possuam capacidades computacionais equivalentes, o que pode ser desafiador em sistemas distribuídos devido à sua complexidade.

5 Tecnologias de TEE para a Implementação de Protocolo de Troca Justa com TTPs Descentralizados.

A segurança nas aplicações computacionais tornou-se cada vez mais importante ao longo dos anos, impulsionando o desenvolvimento de soluções que incorporam mecanismos de segurança no hardware. A Computação Confiável, também conhecida como *Trusted Computing*, é um campo dedicado a aprimorar a segurança e a confiabilidade em sistemas computacionais.

A Computação Confiável envolve tecnologias como o TEE, que cria ambientes isolados e seguros para a execução de aplicativos sensíveis [17]. Exemplos notáveis incluem Intel *SGX, ARM TrustZone, AMD SEV e Morello Board*. Essas tecnologias desempenham um papel fundamental na proteção de dados sensíveis e na garantia da segurança em sistemas e aplicativos diversos.

O *SGX* cria enclaves seguros para a execução de código confidencial, protegendo informações sensíveis contra acessos não autorizados [6]. O *TrustZone* abrange todo o sistema, separando ambiente seguro (mundo confiável) e um ambiente normal (mundo não confiável)[2]. Logo, o *SEV* combina criptografia de memória com a arquitetura de virtualização AMD-V, protegendo máquinas virtuais em ambientes de nuvem [12]. Já o *Morello Board* utiliza a arquitetura *CHERI* para reforçar a segurança de memória, proporcionando resistência a explorações de vulnerabilidades [10].

Frente ao exposto, as tecnologias TEEs são promissoras para resolver o problema da troca justa e outras questões em sistemas distribuídos. Elas fornecem um conjunto de recursos que podem ajudar a tornar o processo de troca justa mais seguro e eficiente, sem a necessidade de uma entidade centralizada para resolver a troca.

6 Protocolos de Troca Justa Baseados em *Attestables*.

Examinamos os TTPs centralizados, destacando a distinção entre protocolos de troca justa fortes e fracos, que dependem dessa terceira parte para garantir a justiça. Em TTPs centralizados, a intrusividade das regras e a implementação monolítica, onde o TTP executa todas as operações internamente, são limitações que buscamos superar. Com as tecnologias TEEs é possível implementar protocolo de troca justa forte, descentralizado e mais seguro sem depender de terceiros confiáveis. A proposta de Avoine [5] abordou a ideia, mas sem o suporte de tecnologias adequadas, utilizando módulos de segurança denominados "anjos"e participantes chamados "piratas", sem o uso de TEEs.

Protocolos de troca justa baseados em *attestables* são uma alternativa viável. Um *attestable* é um modelo conceitual que serve como base para ambientes de execução confiáveis. Esses protocolos transitam de um TTP monolítico com estado para uma versão dividida e sem estado, onde o TTP continua como mediador por meio da combinação de dois componentes: *attestable* (um para cada participante) e o Public Bulletin Board (PBB).

7 Componentes e Operações da Arquitetura Proposta

Os protocolos de troca justa envolvem quatro operações principais, realizadas por dois *attestables* e um PBB, que garantem a justiça da troca quando executadas na ordem correta. Inicialmente, Alice e Bob realizam um *handshake* para estabelecer a comunicação e, em seguida, depositam seus itens. Após isso, ocorre a verificação dos itens, seguida da sincronização, que decide se os itens serão liberados ou restaurados. Se a sincronização for bem-sucedida, o item é liberado; caso contrário, a troca é restaurada.

A operação de depósito ocorre após o handshake, onde Alice e Bob depositam seus itens, garantindo que o outro não tenha acesso a eles ou a capacidade de alterá-los. A operação de verificação assegura que os itens depositados correspondem às descrições acordadas. A operação de sincronização coordena os participantes, garantindo que todos os itens sejam liberados ou restaurados simultaneamente. Finalmente,

a operação de liberação/restaurar decide o destino dos itens, entregando-os ou retornando-os aos donos originais, dependendo do sucesso ou falha da sincronização.

Para realizar essas operações, diferentes ambientes podem ser usados. Ambientes dependentes oferecem controle total a um participante, enquanto ambientes independentes garantem que as operações não possam ser comprometidas por ações inadequadas de um dos participantes. Armazenamento e computação independentes referem-se a ambientes onde a computação e o armazenamento são criptografados e isolados, permitindo maior segurança.

8 Implementação de Protocolo de Troca Justa com TTP Descentralizado que Utiliza TEE para Troca de Itens

Neste artigo, utilizamos um ambiente de computação independente chamado "*attestable*"e um ambiente de mensagens independente denominado "PBB"para implementar nosso protocolo de troca justa. Esses componentes asseguram a execução das operações de troca sem a necessidade de confiar em um terceiro centralizado, aumentando a segurança e a confiabilidade do processo.

O protocolo proposto é projetado para permitir a troca de itens copiáveis, ou seja, dados que podem ser duplicados pelo remetente. Isso garante que Alice não precise devolver o item caso a troca seja cancelada, pois ela nunca entrega o item fisicamente. O processo pode ser interrompido sem prejuízo, desde que Bob não receba o item.

Nos protocolos baseados em "*attestable*", a função do TTP é dividida em dois componentes: "*attestable*"e "PBB", o que reduz a superfície de erro. Em um protocolo descentralizado, dois *attestables*, um para cada participante, garantem a segurança, realizando as operações de ocultação, bloqueio e liberação de itens de forma sincronizada. A troca ocorre em três etapas: Criptografia e Envio dos Documentos, Depósito e Verificação. Na primeira etapa, os documentos são criptografados com a chave compartilhada. No Depósito, os documentos criptografados são trocados entre Alice e Bob. Durante a Verificação, os *attestables* garantem que os documentos correspondem ao esperado, publicando um token de sincronização ou cancelamento no PBB conforme o caso.

Em nossa proposta, Alice possui um att_A e Bob possui um att_B. Estes componentes do TTP fornecem computação independente. O PBB não participa de nenhum processo computacional. Este componente executa a operação sincronização e com base no resultado, os *attestables* (att_A e att_B) decidem se liberar ou restaurar os itens da troca. A execução do protocolo de sincronização é considerada uma das fases mais críticas desses cenários de troca. Isso porque, o protocolo de troca justa dependerá do resultado do protocolo de sincronização para tomar a decisão de Liberar as chaves para os piratas ou Restaurar. Essa decisão é fundamental para o desfecho do protocolo de troca justa, pois determinará se a transação ocorrerá de forma segura e justa.

Nós fazemos as seguintes suposições com base nas funcionalidades e nos blocos de construção do protocolo: D_A e D_B são itens copiáveis; o PBB e os *attestables* dos participantes estão livres de ameaças, falhas ou acidentes; os aplicativos de Alice e Bob podem ser manipulados por seus respectivos proprietários para alterar a computação e a comunicação; os canais $seccha_{AB}$, $seccha_{A2P}$, e $seccha_{B2P}$ não são confiáveis, pois os aplicativos podem atrasar, reordenar ou descartar mensagens; e os *attestables* de Alice e Bob operam de forma independente, comunicando-se apenas via postagem e recebimento de tokens no PBB.

9 Conclusão

Apresentamos uma proposta de protocolo de Troca Justa forte com TTP descentralizado, utilizando tecnologias de TEEs como *Intel SGX*, *TrustZone ARM*, *AMD Secure Memory Encryption* e a *Morello Board*, garantindo a integridade das transações em nível de hardware, a privacidade por meio da divisão do TTP (*attestable* e PBB), e permitindo trocas seguras sem a necessidade de intermediários. A solução oferece maior privacidade e autonomia, com operações de depósito, verificação e liberação/restauração realizadas em ambientes de computação independentes e controlados pelos participantes, enquanto a sincronização ocorre no PBB, assegurando imparcialidade. A modularidade dos *attestables* e PBB garante uma implementação flexível e segura, o que pode promover confiabilidade, privacidade e eficiência nas transações. Trabalhos futuros incluem aprimorar o protocolo com cenários práticos, analisar itens únicos e copiáveis, explorar o uso de *attestables* reais para controlar a troca, focar na sincronização e comparar sincronizações probabilísticas e determinísticas.

DOI: 10.5540/03.2026.012.01.0295

10 Acknowledgments

Esta pesquisa foi parcialmente financiada pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), no âmbito dos projetos 311011/2022-5, 309425/2023-9, 402915/2023-2.

Referências

[1] A. Abadi, S. Murdoch e T. Zacharias. "Recurring contingent service payment". Em: **2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)**. IEEE. 2023, pp. 724–756. DOI: https://Doi.org/10.1109/EuroSP57164.2023.00049.

[2] Tiago Alves. "Trustzone: Integrated hardware and software security". Em: **Information Quarterly** 3 (2004), pp. 18–24. DOI: https://cir.nii.ac.jp/crid/1572824500864199424.

[3] Nadarajah Asokan, Matthias Schunter e Michael Waidner. "Optimistic protocols for fair exchange". Em: **Proceedings of the 4th ACM Conference on Computer and Communications Security**. 1997, pp. 7–17. DOI: https://dl.acm.org/Doi/pdf/10.1145/266420.266426.

[4] Nadarajah Asokan, Victor Shoup e Michael Waidner. "Asynchronous protocols for optimistic fair exchange". Em: **Proceedings. 1998 IEEE Symposium on Security and Privacy (Cat. No. 98CB36186)**. 1998, pp. 86–99. DOI: https://ieeexplore.ieee.org/document/674826.

[5] G. Avoine e S. Vaudenay. "Fair exchange with guardian angels". Em: **Information Security Applications: 4th International Workshop, WISA 2003 Jeju Island, Korea, August 25-27, 2003 Revised Papers 4**. Springer. 2004, pp. 188–202. DOI: https://Doi.org/10.1007/978-3-540-24591-9_15.

[6] V. Costan e S. Devadas. **Intel SGX Explained**. Cryptology ePrint Archive, Paper 2016/086. 2016. DOI: \url{https://eprint.iacr.org/2016/086}. URL: https://eprint.iacr.org/2016/086.

[7] O. Ersoy, Z. Genç, Z. Erkin e M. Conti. "Practical exchange for unique digital goods". Em: **2021 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS)**. IEEE. 2021, pp. 49–58. DOI: https://Doi.org/10.1109/DAPPS52256.2021.00011.

[8] Shimon Even, Oded Goldreich e Abraham Lempel. "A randomized protocol for signing contracts". Em: **Communications of the ACM** 28.6 (1985). Ed. por USA ACM New York NY, pp. 637–647. DOI: https://Doi.org/10.1145/3812.3818.

[9] M. Franklin e M. Reiter. "Fair exchange with a semi-trusted third party". Em: **Proceedings of the 4th ACM Conference on Computer and Communications Security**. 1997, pp. 1–5. DOI: https://Doi.org/10.1145/266420.266424.

[10] R. Grisenthwaite, G. Barnes, R. Watson, S. Moore, P. Sewell e J. Woodruff. "The Arm Morello Evaluation Platform—Validating CHERI-Based Security in a High-Performance System". Em: **IEEE Micro** 43.3 (2023), pp. 50–57. DOI: http://Doi.org/10.1109/MM.2023.3264676.

[11] J. Jang, S. Kong, M. Kim, D. Kim e B. Kang. "Secret: Secure channel between rich execution environment and trusted execution environment." Em: **NDSS**. 2015, pp. 1–15. DOI: https://Doi.org/10.14722/ndss.2015.23189.

[12] David Kaplan, Jeremy Powell e Tom Woller. "AMD memory encryption". Em: **White paper** 13 (2016), p. 12. DOI: http://docs.amd.com/v/u/en-US/memory-encryption-white-paper.

[13] K. Küçük, A. Paverd, A. Martinw, N. Asokan, A. Simpson e R. Ankele. "Exploring the use of Intel SGX for secure many-party applications". Em: **Proceedings of the 1st Workshop on System Software for Trusted Execution**. 2016, pp. 1–6. DOI: https://Doi.org/10.1145/3007788.3007793.

[14] T Moh. "A public key system with signature and master key functions". Em: **Communications in Algebra** 27.5 (1999), pp. 2207–2222. DOI: https://Doi.org/10.1080/00927879908826559.

[15] H. Pagnia, H. Vogt, F. Gärtner e U. Wilhelm. "Solving fair exchange with mobile agents". Em: **International Symposium on Agent Systems and Applications**. Springer. 2000, pp. 57–72. DOI: https://Doi.org/10.1007/978-3-540-45347-5_6.

[16] I. Ray e I. Ray. "An optimistic fair exchange e-commerce protocol with automated dispute resolution". Em: **Electronic Commerce and Web Technologies: First International Conference, EC-Web 2000 London, UK, September 4–6, 2000 Proceedings**. Springer. 2001, pp. 84–93. DOI: https://Doi.org/10.1007/3-540-44463-7_8.

[17] M. Sabt, M. Achemlal e A. Bouabdallah. "Trusted execution environment: what it is, and what it is not". Em: **2015 IEEE Trustcom/BigDataSE/Ispa**. Vol. 1. IEEE. 2015, pp. 57–64. DOI: http://Doi.org/10.1109/Trustcom.2015.357.

[18] Liang Zhang, Haibin Kan, Feiyang Qiu e Feng Hao. "A publicly verifiable optimistic fair exchange protocol using decentralized CP-ABE". Em: **The Computer Journal** 67.3 (2024), pp. 1017–1029. DOI: https://Doi.org/10.1093/comjnl/bxad039.

