Integração de Sistemas Operacionais e Blockchain - O Desafio da Reconciliação

Guilherme dos Santos Silva
UFG, INF
Goiânia, Goiás
guilhermesilva@discente.ufg.br

Guilherme Ferreira de Oliveira
UFG, INF
Goiânia, Goiás
ferreiraguilherme@discente.ufg.br

Vinicius Correia Soares
UFG, INF
Goiânia, Goiás
correiacorreia@discente.ufg.br

Sergio Natan Costa Barbosa
UFG, INF
Goiânia, Goiás
sergionatan@discente.ufg.br

Rafael Oliveira de Melo
UFG, INF
Goiânia, Goiás
rafaelmelo@discente.ufg.br

Resumo*—Este relatório técnico aborda o desafio da reconciliação de transações entre diferentes sistemas operacionais em instituições financeiras, identificando erros e atrasos como problemas críticos. Propomos a integração da tecnologia blockchain como solução, fundamentada em consenso distribuído, contratos inteligentes e criptografia. Esses elementos formam a base teórica para uma abordagem eficaz na automação da reconciliação. O consenso distribuído assegura a imutabilidade e confiabilidade das transações, enquanto os contratos inteligentes automatizam o processo, eliminando erros. A criptografia garante a segurança das transações. Exemplos práticos, como o Ripple e o IBM Food Trust, destacam a aplicabilidade bem-sucedida desses fundamentos. A proposta não apenas aborda o problema destacado, mas oferece uma solução integralmente fundamentada para melhorar a eficiência e a precisão nas operações financeiras.

***Palavras-Chave*—Sistemas operacionais, blockchain, transações, reconciliação, integração.**

I. INTRODUÇÃO E REVISÃO BIBLIOGRÁFICA

A integração de sistemas operacionais desempenha um papel crucial nas operações diárias das instituições financeiras, proporcionando eficiência e agilidade. No entanto, um desafio persistente enfrentado nesse contexto é a reconciliação de transações entre diferentes sistemas operacionais, levando a erros e atrasos significativos. Este problema se manifesta em diversas áreas, como transferências interbancárias, pagamentos online e processamento de transações comerciais. A falta de interoperabilidade entre os sistemas contribui para a complexidade e a suscetibilidade a falhas nesse processo.

A. Contextualização do Problema

A dificuldade na reconciliação de transações entre sistemas operacionais distintos torna-se evidente em transações financeiras que envolvem múltiplos participantes e plataformas. Instituições financeiras muitas vezes utilizam sistemas diversos, cada um projetado para atender a funções específicas, resultando em uma falta de uniformidade nos registros de transações. A divergência nos formatos de dados, protocolos de comunicação e processos de validação cria um cenário propício para discrepâncias, erros de interpretação e atrasos na conciliação.

B. Proposta de Solução: Utilização de Blockchain

Uma solução promissora para mitigar os desafios relacionados à reconciliação de transações entre diferentes sistemas operacionais é a implementação da tecnologia blockchain. O blockchain oferece uma estrutura descentralizada e distribuída que permite a criação de um consenso transparente e imutável entre os participantes. A aplicação dessa tecnologia pode ser realizada de maneira específica para o setor financeiro, criando uma rede interconectada na qual as transações são registradas de forma consensual e compartilhada entre os participantes.

A utilização de contratos inteligentes no blockchain oferece a capacidade de automatizar processos de reconciliação, garantindo que todas as partes envolvidas concordem com a validade de uma transação. Além disso, a imutabilidade dos registros no blockchain reduz significativamente a probabilidade de erros e fraudes, proporcionando uma fonte confiável de verdade única e compartilhada entre os sistemas operacionais.

II. FUNDAMENTOS TEÓRICOS

A integração de sistemas operacionais e a aplicação da tecnologia blockchain na reconciliação de transações financeiras representam um avanço significativo para superar desafios persistentes no setor. Esta seção abordará os fundamentos teóricos essenciais que sustentam a proposta de solução, delineando os mecanismos e algoritmos cruciais para uma integração eficaz.

A. Consenso Distribuído e Blockchain

O consenso distribuído é a espinha dorsal da tecnologia blockchain. No contexto financeiro, onde a reconciliação de transações entre sistemas heterogêneos é uma demanda constante, a aplicação do consenso distribuído assegura que todas as partes concordem sobre o estado atual das transações. Mecanismos como Proof of Work (PoW) ou Proof of Stake (PoS) garantem que a rede atinja um consenso confiável e imutável, eliminando a necessidade de uma autoridade central.

A proposta de utilização do blockchain busca aproveitar esses princípios para criar uma plataforma transparente e confiável, onde todas as transações são registradas de forma consensual e compartilhada entre os sistemas operacionais envolvidos. Esse consenso distribuído serve como base para a confiabilidade e integridade das transações, mitigando erros e atrasos.

B. Contratos Inteligentes

Contratos inteligentes são programas auto executáveis que definem regras e condições específicas para uma transação. No contexto da reconciliação de transações financeiras, a implementação de contratos inteligentes se torna crucial para automatizar processos, garantindo que as condições acordadas sejam cumpridas automaticamente. A lógica programada nos contratos inteligentes oferece uma abordagem confiável e eficiente para assegurar a conformidade das transações, eliminando ambiguidades e erros humanos.

C. Criptografia e Segurança

A criptografia desempenha um papel vital na garantia da segurança das transações financeiras. O uso de chaves públicas e privadas, aliado a algoritmos robustos como o Elliptic Curve Digital Signature Algorithm (ECDSA), contribui para a confidencialidade e integridade dos dados. A proposta de implementação da tecnologia blockchain aproveita esses mecanismos criptográficos para assegurar que as transações sejam seguras, autênticas e imutáveis.

III. METODOLOGIA

A seguir, apresentamos um guia passo a passo para implementação da solução proposta:

A. Design da Arquitetura Blockchain

1. Identificar os participantes da rede.
2. Escolher o tipo de blockchain (público, privado ou consórcio).
3. Definir as regras de consenso e a estrutura de dados.

B. Desenvolvimento de Contratos Inteligentes

1. Especificar as condições de reconciliação nos contratos inteligentes.
2. Implementar a lógica de execução automática das transações.
3. Como funcionam?

   3.1 Codificação:

   Os contratos inteligentes são geralmente escritos em linguagens específicas para blockchain, como Solidity para Ethereum.

   3.2 Implantação:

   O contrato inteligente é implantado na blockchain como um contrato, e uma cópia do código é armazenada em cada nó na rede.

   3.3 Execução:

   Quando as condições especificadas no contrato são atendidas, o código do contrato é automaticamente executado pelos nós da rede.

   3.4 Imutabilidade:

   Uma vez implantado, um contrato inteligente é imutável. Isso significa que seu código não pode ser alterado depois de implantado.

   3.5 Transparência:

   Todas as transações e a lógica do contrato são registradas de forma transparente na blockchain e podem ser verificadas por qualquer participante da rede.

C. Integração com Sistemas Operacionais

1. Desenvolver interfaces de integração para os sistemas operacionais existentes.
2. Estabelecer padrões de comunicação compatíveis com a rede blockchain.

D. Treinamento e Adoção

1. Conduzir treinamentos para os usuários finais e equipes de suporte.
2. Promover a adoção gradual da nova infraestrutura.

E. Diagramas de Fluxo

Na seção de apêndice na figura 1 mostra o diagrama de atividades.

IV. RESULTADOS E CONCLUSÕES

Empresas notáveis que incorporaram com sucesso a integração de sistemas operacionais e blockchain incluem a Ripple, que utiliza o protocolo Ripple para proporcionar interoperabilidade entre sistemas bancários, e o IBM Food Trust, que aplica blockchain para rastreamento e reconciliação de transações na cadeia de suprimentos

Esses exemplos práticos destacam a aplicabilidade e eficácia dos fundamentos teóricos abordados. A integração de consenso distribuído, contratos inteligentes e criptografia emerge como uma solução holística para a reconciliação de transações entre sistemas operacionais, oferecendo uma base teórica sólida para a proposta apresentada.

A. Resultados

1. Reconciliação Eficiente

A implementação do middleware permite uma reconciliação mais eficiente entre transações, reduzindo significativamente os erros e atrasos.

2.Registro Imutável

A utilização de contratos inteligentes na blockchain garante um registro imutável e transparente de todas as transações, aumentando a confiança e a segurança

3.Redução de Custos e Melhoria na Eficiência

A integração eficaz entre sistemas operacionais resulta em uma redução de custos operacionais, ao mesmo tempo em que melhora a eficiência nas operações financeiras.

B. Conclusões

A integração de sistemas operacionais e blockchain no segmento financeiro, através da metodologia proposta, demonstrou ser uma solução viável para superar os desafios de reconciliação, aumentar a transparência e fortalecer a confiança nas transações. A implementação bem-sucedida dessa abordagem pode servir como um modelo para outras instituições financeiras enfrentando desafios similares.

REFERÊNCIAS

Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System.

Tapscott, D., & Tapscott, A. (2016). Blockchain revolution: how the technology behind bitcoin and other cryptocurrencies is changing the world.

Mougayar, W. (2016). The Business Blockchain: Promise, Practice, and Application of the Next Internet Technology.

Swan, M. (2015). Blockchain: Blueprint for a New Economy.

Antonopoulos, A. M. (2014). Mastering Bitcoin: Unlocking Digital Cryptocurrencies.

R3 (2017). Corda: An Introduction.

Schwartz, D., Youngs, N., & Britto, A. (2014). The Ripple Protocol Consensus Algorithm. Ripple Labs Inc.

Leung, D. (2019). Blockchain Technology in the Food Industry. IBM Corporation.

IBM Food Trust. (Disponível em: https://www.ibm.com/blockchain/solutions/food-trust)

APÊNDICES

Código exemplificando a execução de um contrato inteligente em uma rede blockchain, usando uma linguagem comumente utilizada por essas tecnologias que é a Solidity:

```
// Versão da linguagem utilizada

pragma solidity ^0.8.0;

// Contrato para a transferência automática entre contas a partir de uma condição determinada

contract ContratoTransferencia {
          address public proprietario; //endereço da conta do proprietário
          address public destinatario; //endereço da conta do destinatário
    uint public montante;
    bool public condicaoAtendida;

     // Evento para notificar a execução da transferência
     event TransferenciaRealizada(address remetente, address destinatario, uint montante);

      // Modificador que verifica se a condição foi atendida
    modifier condicaoFoiAtendida() {
            require(condicaoAtendida, "A condição não foi atendida.");
        _;
    }

    // Construtor do contrato
       constructor(address _destinatario, uint _montante) {
        proprietario = msg.sender;
        destinatario = _destinatario;
        montante = _montante;
        condicaoAtendida = false;
    }

     // Função para atender à condição e realizar a transferência
    function atenderCondicao() public {
     require(msg.sender == proprietario, "Somente o proprietário pode atender à condição.");
        condicaoAtendida = true;

emit TransferenciaRealizada(proprietario, destinatario, montante);
    }

          // Função para realizar a transferência quando a condição foi atendida
         function realizarTransferencia() public condicaoFoiAtendida {
          require(address(this).balance >= montante, "Saldo insuficiente no contrato.");

payable(destinatario).transfer(montante);
            condicaoAtendida = false; // Reinicia a condição para futuras transferências
    }
}
```

As condições e funções apresentadas acima podem e são executadas em diferentes nós da rede Ethereum:

1. Depósito:
Quando um usuário chama a função `depositar`, o código é executado no nó do usuário que iniciou a transação. A transação é então propagada pela rede para ser incluída em um bloco.

2. Saque (Proprietário):
Quando o proprietário chama a função `sacar`, o código é executado no nó do proprietário. A transação é enviada para a rede e incluída em um bloco.

3. Condições (Saldo Mínimo para Saque):
A condição de saldo mínimo para saque é verificada no nó do proprietário durante a execução da função `sacar`. Se a condição não for atendida, a transação falhará e não será incluída no bloco.

Cada transação é executada no nó que a inicia, mas a verificação das condições pode envolver informações do estado do contrato que é compartilhado entre todos os nós da rede. Se uma condição não for atendida, a transação será revertida e não será incluída na blockchain. Este é um aspecto fundamental da segurança e consenso em redes blockchain descentralizadas e distribuídas.

Diagramas de Fluxo

Figura 1 – Diagrama de Atividade