Avaliação de Desempenho de Rede *Hyperledger Fabric* CA para Registro de Presença em Eventos ao Ar Livre

Marco A. C. Silva[1,2], Luis H. V. Nakamura[2], Geraldo P. Rocha Filho[3], Rodolfo I. Meguette[1]

[1]Instituto de Computação e Ciência Computacional (ICMC)
Universidade de São Paulo (USP)
Av. Trabalhador São-carlense, 400 – 13566-590 – São Carlos – SP – Brasil

[2]Instituto Federal de Ciência, Educação e Tecnologia de São Paulo (IFSP)
Av. Pastor José Dutra de Moraes, 239 – 15808-305 – Catanduva – SP – Brasil

[3]Universidade Estadual do Sudoeste da Bahia (UESB)
Estr. Bem Querer, Km-04, 45083-900 – Vitória da Conquista – BA – Brasil

marco.colombo@usp.br, nakamura@ifsp.edu.br,

geraldo.rocha@uesb.edu.br, meneguette@icmc.usp.br

Abstract. *With the advancement of technologies for recording data in distributed networks, the concern of users and developers of computerized solutions with the privacy of sensitive data has increased. To address this topic in the blockchain networking environment, precisely as planned in the Hyperledger project environment, some distributed ledger solutions are available, such as Hyperledger Fabric with Hyperledger Caliper. This article aims to compare the performance of two Hyperledger Fabric networks. To achieve this, we carried out a controlled experiment where both networks operate a smart contract that manages attendance records at outdoor events. The main difference between the networks is that one uses a Certificate Authority (CA) to issue access certificates, while the other performs manual issuance of certificates. We compared the results obtained through reports from the Hyperledger Caliper tool, also available in the Hyperledger project ecosystem. The results of this study provide valuable information that can help developers choose the most suitable ledger type for their Hyperledger projects.*

Resumo. *Com a avanço das tecnologias para registro de dados em rede distribuídas, a preocupação de usuários e desenvolvedores de soluções informatizadas com a privacidade de dados sensíveis tem aumentado. Para o tratamento deste tópico no ambiente de redes blockchain, especificamente as desenvolvidas no ambiente do projeto Hyperledger, estão disponíveis algumas soluções de livros-razão distribuídos, como o Hyperledger Fabric com o Hyperledger Caliper. O objetivo deste artigo é comparar o desempenho de duas redes Hyperledger Fabric. Para isso, realizamos um experimento controlado onde ambas as redes operam um contrato inteligente que gerencia registros de presença em eventos ao ar livre. A diferença principal entre as redes é que uma utiliza uma Certificate Authority (CA) para emitir certificados de acesso, enquanto a outra realiza a emissão manual de certificados. A comparação é feita com os resultados obtidos através dos relatórios da ferramenta Hyperledger Caliper, também*

disponível no ecossistema do projeto Hyperledger. Os resultados deste estudo oferecem informações valiosas que podem ajudar desenvolvedores a escolher o tipo de livro-razão mais adequado para seus projetos Hyperledger.

1. Introdução

Múltiplas aplicações pertencentes a entidades privadas e agências governamentais exigem a manipulação de informações sensíveis de maneira meticulosa e segura. Independentemente do setor, é imperativo o estabelecimento e a observância rigorosa de normativas claras de privacidade pelas corporações envolvidas.

Paralelamente, os usuários, cuja segurança depende do estrito cumprimento dessas normas, devem não apenas estar cientes de tais diretrizes mas também ser assegurados quanto à sua efetiva implementação em face do aumento de esforços mal intencionados para quebra de sigilo de dados [Bu et al. 2020].

Atualmente, observa-se que a maioria das identidades digitais dos usuários é centralizada e gerenciada por um restrito número de grandes corporações, privando os usuários do controle sobre suas próprias informações pessoais [Liu et al. 2020]. Este cenário contribui para uma comercialização de dados caracterizada por uma notável falta de transparência [Lux et al. 2019, Maschi et al. 2018].

Existem abordagens e instrumentos tecnológicos que possibilitam a realização de autenticação anônima de usuários em sistemas computacionais descentralizados de maneira segura e confiável, ao mesmo tempo que proporcionam a personalização da gestão de dados sensíveis. Essas ferramentas fazem uso de *blockchains*, os quais representam livros-razão distribuídos, completamente acessíveis ao público e reconhecidos por sua elevada segurança [Carlozo 2017].

As características de privacidade do *blockchain* são cruciais para aumentar a confiabilidade associada ao uso desta tecnologia e para impulsionar inovações em técnicas defensivas e de contramedidas. [Zhang et al. 2019].

Dentro deste contexto, este trabalho tem por objetivo a avaliação do desempenho de duas redes *blockchain*, sendo uma delas implementada com uma gestão de certificados digitais sofisticada e completa e a outra com gestão manual de certificados digitais.

As informações aferidas com base nos resultados dos testes deste trabalho poderão ser levadas em consideração por desenvolvedores no diz respeito a adoção ou não da *Certificate Authority* (CA) de acordo com o volume de certificados e entidades que suas aplicações esperam gerir.

A organização deste texto está de acordo com as seguintes seções: A Secão 2 discorre sobre as informações relacionadas ao *blockchain* que são importantes para a compreensão deste trabalho. Os trabalhos da literatura relacionados à este artigo estão na Seção 3. Na Seção 4 apresentamos o experimento de comparação de desempenho realizado. Os resultados obtidos com o experimento são apresentados na Seção 5 e na Seção 6 há uma discussão sobre os resultados. Finalmente, na Seção 7 são apresentadas as nossas considerações finais sobre este trabalho.

2. Blockchain

Da acordo com [Androulaki et al. 2018]:

> "Um *blockchain* pode ser definido como um livro razão imutável para gravação de transações, mantidas dentro de uma rede distribuída entre pares mutuamente desconfiados. Cada par mantém uma cópia do livro-razão. Os pares executam um protocolo de consenso para validar transações, agrupá-las em blocos e construir uma cadeia."

O *Hyperledger Fabric* é uma plataforma de *blockchain* de código aberto, desenvolvida sob o guarda-chuva da Linux Foundation e permite a criação de redes *blockchain* permissionadas, nas quais os participantes são previamente autorizados e cada um pode ter diferentes níveis de acesso e permissões [Foundation 2021b]. De acordo com [Vukoli 2017] o *Hyperledger Fabric* fornece arquitetura modular e inclui componente de associação permitindo que os desenvolvedores criem soluções que se adequam exatamente às necessidades específicas de suas organizações.

O *Hyperledger Fabric* também suporta a execução de "*chaincodes*", conhecidos em outros contextos de *blockchain* como contratos inteligentes, que são *scripts* executados na *blockchain* e que automatizam, validam ou executam negociações de forma segura e eficiente [Foundation 2021b].

Na arquitetura do *Hyperledger Fabric*, a *Certificate Authority* (CA) desempenha um papel central na gestão de identidades, emitindo certificados digitais que são fundamentais para a autenticação e autorização dentro da rede *blockchain*. Esses certificados validam a identidade dos participantes e possibilitam operações seguras e confiáveis dentro da rede [Foundation 2021b]. A presença de uma Autoridade Certificadora no *Hyperledger Fabric* oferece maior segurança, escalabilidade e flexibilidade na gestão de certificados e identidades, tornando-o adequado para ambientes de produção [Gayathri Santhosh and Reshmi 2023].

Já o *Hyperledger Caliper* é uma ferramenta de *benchmarking* de *blockchain* que faz parte do projeto *Hyperledger*, hospedado pela *Linux Foundation*. Seu principal objetivo é medir o desempenho de uma rede *blockchain* específica com várias métricas, como taxa de transações por segundo, latência de transação, uso de recursos (CPU, memória, etc.), e vazão sob diferentes condições de rede e cargas de transação [Foundation 2022].

O *Hyperledger Caliper* suporta múltiplas plataformas de *blockchain*, incluindo *Hyperledger Fabric*, *Hyperledger Sawtooth*, e outras, possibilitando aos usuários testar e comparar o desempenho de diferentes tecnologias de blockchain com um conjunto comum de benchmarks. Vários relatórios com indicadores de desempenho são produzidos pelo *Hyperledger Caliper* [Foundation 2022].

De acordo com o site do projeto *Hyperledger* [Foundation 2021a], a comunidade inclui "líderes em finanças, bancos, internet das coisas, cadeias de suprimentos, manufatura e tecnologia". É importante salientar que as soluções produzidas pelo projeto são de código aberto e sob governança técnica aberta.

3. Trabalhos Correlatos

Encontramos artigos na literatura que realizam testes e avaliações de desempenho em redes *Hyperledger Fabric* considerando diferentes cenários e focando em alvos específicos

em suas análises.

O artigo [Kuzlu et al. 2019] avalia o impacto da carga de trabalho da rede no desempenho de uma plataforma *blockchain* (*Hyperledger Fabric*). Os autores utilizaram o *Hyperledger Caliper* para avaliação de desempenho em termos de vazão, latência e escalabilidade. Os experimentos foram executados no AWS EC2, e os autores concluíram que a vazão, a latência e a escalabilidade de uma rede *blockchain* dependem da configuração do *hardware*, do *design* da rede *blockchain* e da complexidade das operações do contrato inteligente.

Por sua vez, o artigo [Wang and Chu 2020] faz uma avaliação de desempenho da arquitetura de executar-ordenar-validar do *Hyperledger Fabric* revelou que a fase de execução apresentou boa escalabilidade sob a política de endosso OR, com os serviços de ordenação (Solo, Kafka e Raft) apresentando desempenho relativamente bom, e a fase de validação sendo provavelmente o gargalo do sistema devido à baixa velocidade de validação do *chaincode*.

Já o artigo [Melo et al. 2022] avaliou o desempenho de uma plataforma *Hyperledger Fabric* (v1.4.1) implantada em um ambiente privado. Uma única entidade foi gerenciada e avaliada (latência e vazão) usando a ferramenta de *benchmark Caliper*. Os autores detectaram o aumento no consumo de recursos e encontraram um problema relacionado ao envelhecimento do software. O artigo também inclui uma avaliação interessante do consumo de recursos computacionais, como CPU, RAM, Disco e memória cache. Assim, os autores criaram um modelo de disponibilidade do sistema considerando um aumento no consumo desses recursos e revelando o impacto na disponibilidade geral do sistema.

Finalmente, no artigo [Mor et al. 2024] os autores avaliam e comparam o desempenho de diferentes versões do *Hyperledger Fabric* (v1.0 até v1.4.4), destacando que, sob circunstâncias de alta carga de trabalho, o desempenho da plataforma não correspondeu ao dos sistemas de banco de dados convencionais contemporâneos. Os autores esperam que os resultados deste estudo ajudem a comunidade corporativa a selecionar a melhor plataforma *blockchain* para suas necessidades.

Tabela 1. Características de interesse dos Trabalhos Correlatos

Trabalho	Comparação entre redes	Utiliza Caliper	Foco na CA
[Kuzlu et al. 2019]	Não	Sim	Não
[Wang and Chu 2020]	Sim	Não	Não
[Melo et al. 2022]	Não	Sim	Não
[Mor et al. 2024]	Sim	Não	Não
Este Trabalho	Sim	Sim	Sim

Como pode ser percebido pela Tabela 1, apesar dos trabalhos supramencionados utilizarem ferramentas correlatas para a execução de avaliação de desempenho em redes *Hyperledger* e cumprirem seus objetivos em variados contextos, diferentemente deste trabalho, não foram concebidos especificamente para a avaliação de desempenho e comparação de redes com o *Certificate Authority* (CA) e redes que não contam com este gestor de certificados digitais.

4. Experimento

O cenário escolhido para implementação de instâncias de livro-razão utilizando os métodos de Gravação e Leitura foi o registro de participação em eventos urbanos ao ar livre, como por exemplo paradas, desfiles, manifestações, de forma controlada e, ao mesmo tempo, preservando a identidade dos participantes com registro confiável.

O contrato-inteligente, o *benchmarks*, as cargas de tabalho, assim como tabelas com os resultados deste trabalho estão disponíveis em: https://github.com/macs20/wperformance2024.

O registro de evento ao ar livre contou com os seguintes atributos:

- **Id do usuário** - informação que representa a identificação do usuário de um sistema de registro de presença em eventos ao ar livre. Para simular a escolha de um ID, foi gerado um número pseudorrandômico entre 1 e 5000.
- **Tipo Evento** - O tipo de evento ao livre cuja presença está sendo registrada no *blockchain*. Para simular a escolha do evento do qual o usuário estaria presente, foi sorteado um tipo de evento em um vetor pré-determinado.
- **Latitude** - Representa o primeiro elemento para a formar a coordenada necessária para a estabelecer a localização do usuário no momento do registro de presença. Foi gerado um valor válido para latitude de forma pseudorrandômica.
- **Longitude** - Representa o segundo elemento para a formar a coordenada. Foi gerado um valor válido para longitude de forma pseudorrandômica.
- **Data** - A data completa do momento em que o registro de presença foi realizado, incluindo o horário. Um valor válido de data completa foi regerado de forma pseudorrandômica.

4.1. Métricas

O *Hyperledger Caliper* avalia o desempenho de sistemas de *blockchain* utilizando várias métricas chave, que proporcionam uma compreensão abrangente da eficiência, escalabilidade e robustez de uma rede. As métricas utilizadas neste trabalho são:

- **Taxa de Transações por Segundo (TPS)**: Mede o número de transações que a rede é capaz de processar por segundo. É uma métrica importante para entender a capacidade de vazão da rede.
- **Latência da Transação**: Refere-se ao tempo necessário para que uma transação seja confirmada pela rede *blockchain*. Inclui o tempo desde a submissão da transação até a sua inclusão em um bloco e a eventual confirmação desse bloco pela rede.
- **Uso de Recursos**: Inclui o monitoramento do consumo de recursos de computação pela rede *blockchain*, como uso de CPU e memória. Essa métrica é vital para avaliar a eficiência do sistema e identificar potenciais gargalos de desempenho.
- **Vazão**: A capacidade total da rede de processar transações em um período específico. Difere da taxa de transações por segundo, pois pode considerar também a capacidade da rede de lidar com cargas de trabalho variáveis ao longo do tempo.

A utilização dessas métricas objetiva fornecer aos desenvolvedores e administradores de rede visões sobre o desempenho operacional e a eficiência de uma implantação de *blockchain*, permitindo-lhes fazer ajustes otimizados para melhorar a capacidade e a confiabilidade da rede.

4.2. Especificações

O teste foi executado em um ambiente virtual com o sistema operacional Ubuntu 20.04 em um computador Dell G7 com 16GB de memória RAM, processador Intel Core i7 de oitava geração.

A versão escolhida do *Hyperledger Fabric* foi a 2.5.6 e a versão do *Hyperledger Caliper* foi a 0.4.1. Uma rede foi implemantada com o *Cetificate Autority (CA)* e a outra com criação e gestão manual de ceritificados.

Neste experimento o *chaincode* foi escrito em *Javascript* e as operações de leitura e gravação foram executadas separadamente em cada uma das redes.

A carga de trabalho e a coleta de métricas de avaliação foram geradas pelo *Hyperledger Caliper* e a faixa de valores nele configurado para a taxa de envio das requisições entre 50 e 200 transações por segundo.

Foram realizadas 3 execuções do teste, sendo a primeira com 1.000 transações, a segunda com 5.000 transações e a terceira com 10.000 transações.

5. Resultados

Após a execução dos testes com as redes *Hyperledger Fabric*, foram obtidos os resultados apresentados nesta seção. Para a diferenciação entre entre as duas redes, nos referiremos a rede que implementa o *Certiticate Authority* como Rede CA e Rede MSP a rede que não o faz.

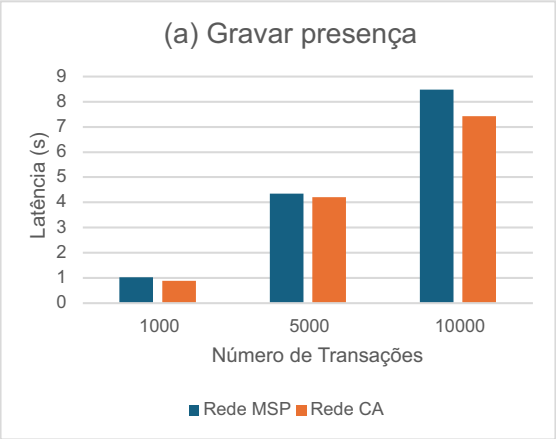

Figura 1. Resultados da Latência para o método Gravar presença.

Com relação ao teste de latência, para a função de gravação de uma presença, conforme pode ser observado no gráfico da Figura 1, como resultado da execução com 1.000 transações, a Rede MSP obteve uma latência de 1,03 segundos, a Rede CA obteve uma latência de 0,88 segundos.

Para a execução com 5.000 transações, foram registradas as latências de 4,34 segundos para a Rede MSP e 4,2 segundos para Rede CA.

Já para a execução com 10.000 transações, a Rede MSP obteve uma latência de 8,48 segundos e a Rede CA uma latência de 7,43 segundos. Quando a função de leitura de um registro de presença foi executada, conforme pode ser observado no gráfico (b) da

Figura 2, como resultado de todas as execuções, o resultado foi 0,01 segundos para as duas redes.

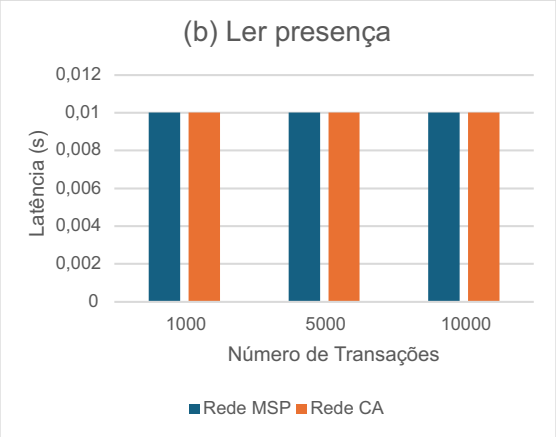

Figura 2. Resultados da Latência para o método Ler presença.

No que se refere à vazão, para a função de gravação de uma presença, conforme pode ser observado no gráfico da Figura 3, como resultado da execução com 1.000 transações, a Rede MSP obteve uma vazão de 46,1 transações por segundo, a Rede CA obteve uma vazão de 45,3 transações por segundo.

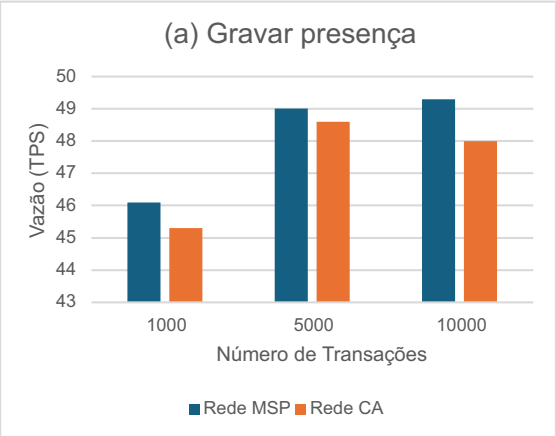

Figura 3. Resultados métrica da Vazão para o método Gravar presença.

Para a execução com 5.000 transações, foram registradas as vazões de 49 transações por segundo para a Rede MSP e 48,6 transações por segundo para Rede CA.

Já para a execução com 10.000 transações, a Rede MSP obteve uma vazão de 49,3 transações por segundos e a Rede CA uma vazão de 48 transações por segundo.

Quando a função de leitura de um registro de presença foi executada, conforme pode ser observado no gráfico da Figura 4, como resultado da execução com 1.000 transações, a Rede MSP obteve uma vazão de 61,2 transações por segundo, a Rede CA obteve uma vazão de 64,1 transações por segundo.

Para a execução com 5.000 transações, foram registradas as vazões de 64,9 transações por segundo para a Rede MSP e 64,8 transações por segundo para Rede CA.

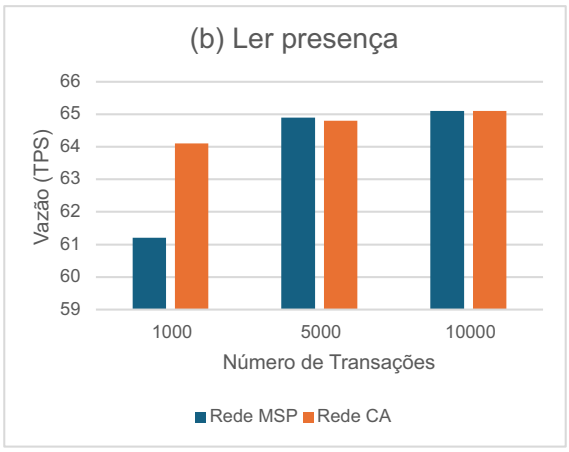

Figura 4. Resultados da Vazão para o método Ler presença.

Já para a execução com 10.000 transações, a Rede MSP obteve uma vazão de 65,1 transações por segundos e a Rede CA uma vazão de 65,1 transações por segundo.

Na aferição do uso de CPU, para a função de gravação de uma presença, conforme pode ser observado no gráfico da Figura 5, como resultado da execução com 1.000 transações, a Rede MSP utilizou 14,79% da capacidade da CPU, enquanto a Rede CA utilizou 15,32% da capacidade da CPU.

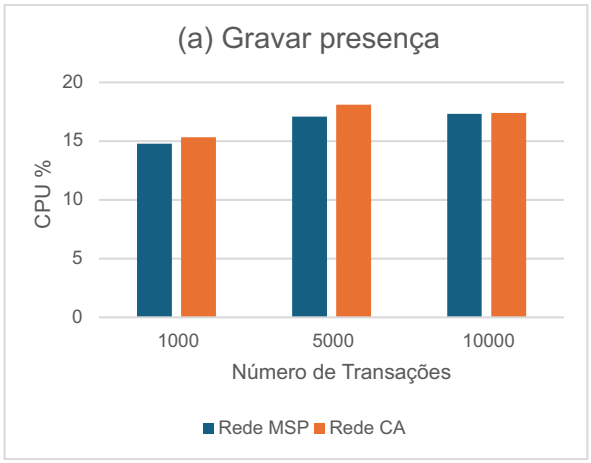

Figura 5. Resultados do uso de CPU para o método Gravar presença.

Para a execução com 5.000 transações, a Rede MSP utilizou 17,1% da capacidade da CPU, enquanto a Rede CA utilizou 18,12% da capacidade da CPU.

Já para a execução com 10.000 transações, a Rede MSP utilizou 17,33% da capacidade da CPU, enquanto a Rede CA utilizou 17,39% da capacidade da CPU.

Quando a função de leitura de um registro de presença foi executada, conforme pode ser observado no gráfico da Figura 6, como resultado da execução com 1.000 transações, a Rede MSP utilizou 8,38% da capacidade da CPU, enquanto a Rede CA utilizou 8,43% da capacidade da CPU.

Para a execução com 5.000 transações, a Rede MSP utilizou 15,87% da capaci-

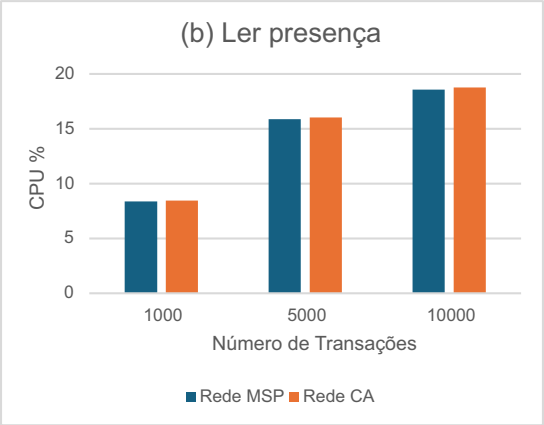

Figura 6. Resultados do uso de CPU para o método Ler presença.

dade da CPU, enquanto a Rede CA utilizou 16,04% da capacidade da CPU. Já para a execução com 10.000 transações, a Rede MSP utilizou 18,55% da capacidade da CPU, enquanto a Rede CA utilizou 18,76% da capacidade da CPU.

Quando aferido o uso de mémoria, para a função de gravação de uma presença, conforme pode ser observado no gráfico da Figura 7, como resultado da execução com 1.000 transações, a Rede MSP utilizou 259 megabytes de memória, enquanto a Rede CA utilizou 172 megabytes de memória.

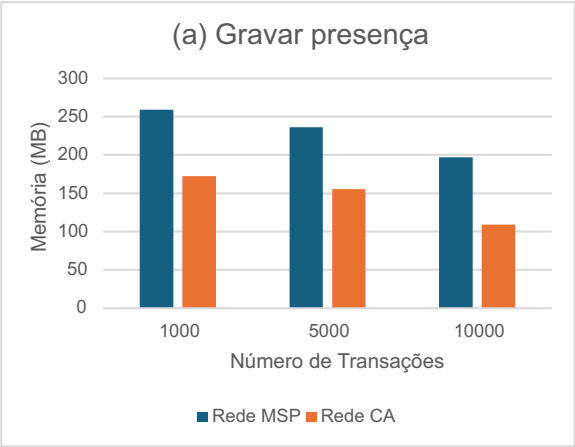

Figura 7. Resultados do uso de Memória para o método Gravar presença.

Para a execução com 5.000 transações, a Rede MSP utilizou 236 megabytes de memória, enquanto a Rede CA utilizou 155 megabytes de memória.

Já para a execução com 10.000 transações, a Rede MSP utilizou 197 megabytes de memória, enquanto a Rede CA utilizou 109 megabytes de memória.

Quando a função de leitura de um registro de presença foi executada, conforme pode ser observado no gráfico da Figura 8, como resultado da execução com 1.000 transações, a Rede MSP utilizou 246 megabytes de memória, enquanto a Rede CA utilizou 72,9 megabytes de memória. Para a execução com 5.000 transações, a Rede MSP utilizou 251 megabytes de memória, enquanto a Rede CA utilizou 71,5 megabytes de

memória. Já para a execução com 10.000 transações, a Rede MSP utilizou 249 megabytes de memória, enquanto a Rede CA utilizou 68,1 megabytes de memória.

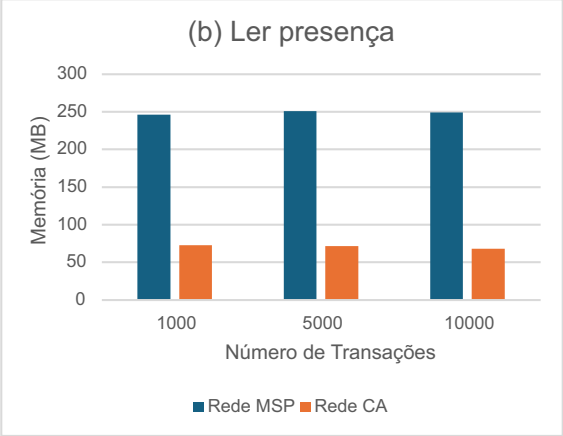

Figura 8. Resultados do uso de Memória para o método Ler presença.

6. Discussão

Com base nos resultados obtidos nos testes, é possível observar que durante o ato de gravar uma presença, os valores de latência em segundos obtidos pela Rede CA foram ligeiramente menores em todas as execuções. Também pode-se notar que a latência aumentou gradualmente quanto maior o número de transações enviadas no teste. O mesmo não ocorreu durante a leitura de presenças, onde todos os valores de latência se mostraram próximos para qualquer quantidade de transações.

Quando analisados os dados obtidos com teste vazão, pode-se oberservar que para a função Gravar presença, a Rede MSP atingiu valores superiores em todos em todas as execuções. No que se refere a esta mesma função, os valores de transções por segundo aumentaram na Rede MSP quando o número de transações foi aumentado, sendo o aumento de 5.000 para 10.000 transações mais discreto que de 1.000 transações para 5.000 transações. O mesmo não aconteceu com a Rede CA que teve uma pequena redução no valor de vazão quando o números de transações aumentou de 5.000 para 10.000.

Observando os dados obtidos com os testes de vazão para a função Ler presença, a diferença mais significativa de valores ocorreu entre a leitura de presenças das duas redes quando o número de transações era de apenas 1.000, visto que nos testes com 5.000 e 10.000 os valores se mantiveram próximos para as duas redes.

Ao observamos os dados sobre a utilização de CPU, é possível notar que, quando executada a função Gravar presença, os valores são ligeiramente maiores na Rede CA do que os valores obtidos na Rede MSP, a não ser pela execução com 10.000 transações que registrou valores semelhantes. Também é possível notar que com o aumento do número de transações, esta função não apresentam grande distância entre si. Diferentemente disto, os valores de uso de CPU da função leitura, onde pode ser observado, primeiramente o aumento gradual do uso de CPU conforme o número de transações aumenta, e também a proximidade dos valores obtidos entre as duas redes.

Com relação aos dados gerados pelo teste de uso de memória, é possível perceber que, ao ser utilizada a função Gravar presença, a Rede CA utiliza menos memória em

todas as execuções. Também faz-se necessário chamar a atenção para o fato de que o uso de memória diminui conforme o número de transações sobe. O mesmo não ocorre nas execuções com a função Ler presença, onde o uso de memória se manteve aproximadamente o mesmo para todas as execuções. Porém, a utilização de memória para esta função é significativamente menor na Rede CA.

Portanto, ao escolher adotar ou não a CA em seu projeto o desenvolvedor deve atentar para dois fatores: (a) o volume de identidade e certificados a serem geridos pela aplicação e (b) a frequencia com que serão executados os métodos de leitura e gravação.

7. Conclusão

O desenvolvimento de redes com o uso do *Certificate Authority* (CA), estrutura crucial para gestão de certificados na rede, deve ser considerado pelos desenvolvedores quando a aplicação requerer uma gestão de certificados sofisticada. Para quando este não for o caso e a natureza da aplicação permitir uma gestão de certificados simplificada, este trabalho obteve resultados que podem servir como subsídios para a decisão sobre a adoção do CA.

Para isto, foi feita a comparação do desempenho de uma rede criada com a CA e uma rede sem esta implementação com os certificados emitidos e distribuídos manualmente.

Os testes mostraram resultados próximos para as métricas: latência, vazão e uso de CPU, porém para métrica Uso de Memória, a rede implementada com CA teve desempenho superior.

Em trabalhos futuros serão realizados testes com uma infraestrutura de hardware de maior capacidade computacional, um maior número de transações e repetições para estabelecer um intervalo de confiança. Também serão considerados outros emissores de certificados e com a utilização de outros tipos de bases de dados para a criação de transações. Avanços nesta na linha investigativa deste trabalho incluem, ainda, uma comparação quantitativa com outras plataformas de *blockchain*.

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001. Rodolfo Meneguette gostaria de agradecer a Fundação de Amparo à Pesquisa do Estado de São Paulo (número 2022/00660-0 e 2020/07162-0) pelo suporte financeiro.

Referências

Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y., et al. (2018). Hyperledger fabric: a distributed operating system for permissioned blockchains. In *Proceedings of the thirteenth EuroSys conference*, pages 1–15.

Bu, F., Wang, N., Jiang, B., and Liang, H. (2020). "privacy by design" implementation: Information system engineers' perspective. *International journal of information management*, 53:102124.

Carlozo, L. (2017). What is blockchain? *Journal of Accountancy*, 224(1):29.

Foundation, L. (2021a). Hyperledger - open soucer blockchain technologies.

Foundation, L. (2021b). Hyperledger fabric- hyperledger foundation.

Foundation, L. (2022). Hyperledger caliper- hyperledger foundation.

Gayathri Santhosh, M. and Reshmi, T. (2023). Enhancing pki security in hyperledger fabric with an indigenous certificate authority. In *2023 IEEE International Conference on Public Key Infrastructure and its Applications (PKIA)*, pages 1–5.

Kuzlu, M., Pipattanasomporn, M., Gurses, L., and Rahman, S. (2019). Performance analysis of a hyperledger fabric blockchain framework: Throughput, latency and scalability. In *2019 IEEE International Conference on Blockchain (Blockchain)*, pages 536–540.

Liu, Y., Lu, Q., Paik, H.-Y., Xu, X., Chen, S., and Zhu, L. (2020). Design pattern as a service for blockchain-based self-sovereign identity. *IEEE software*, 37(5):30–36.

Lux, Z. A., Beierle, F., Zickau, S., and Göndör, S. (2019). Full-text search for verifiable credential metadata on distributed ledgers. In *2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, pages 519–528. IEEE.

Maschi, L. F. C., Pinto, A. S. R., Meneguette, R. I., and Baldassin, A. (2018). Data summarization in the node by parameters (dsnp): Local data fusion in an iot environment. *Sensors*, 18(3).

Melo, C., Oliveira, F., Dantas, J., Araujo, J., Pereira, P., Maciel, R., and Maciel, P. (2022). Performance and availability evaluation of the blockchain platform hyperledger fabric. *The Journal of Supercomputing*, 78(1):12505–12527.

Mor, P., Tyagi, R. K., Jain, C., and Verma, D. K. (2024). Enhanced hyperledger fabric network set-up for remittance and settlement process. In Goyal, S. K., Palwalia, D. K., Tiwari, R., and Gupta, Y., editors, *Flexible Electronics for Electric Vehicles*, pages 157–167, Singapore. Springer Nature Singapore.

Vukoli, M. (2017). Rethinking permissioned blockchains [c]. In *ACM Workshop. ACM*.

Wang, C. and Chu, X. (2020). Performance characterization and bottleneck analysis of hyperledger fabric. In *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*, pages 1281–1286.

Zhang, R., Xue, R., and Liu, L. (2019). Security and privacy on blockchain. *ACM computing surveys*, 52(3):1–34.