ContAudIT: Uma Proposta Fim-a-Fim para Auditoria Contínua de Gerência de Mudanças em TI usando *Blockchain*

Carlos Fraga[1], **Antônio Abelém**[2], **Vinícius Borges**[3], **Jeferson Nobre**[1], **Juliano Wickboldt**[1], **Glauber Goncalves**[4], **Billy Pinheiro**[5], **Weverton Cordeiro**[1]

[1] Universidade Federal do Rio Grande do Sul, [2] Universidade Federal do Pará, [3] Universidade Federal de Goiás, [4] Universidade Federal do Piauí, [5] Amazônia Blockchain Solutions

Abstract. *IT changes are a critical part of the day-to-day operations of most modern organizations, and poor change management can pose severe risks to business continuity. In this context, shareholders often resort to auditing to ensure change management following accredited procedures. To this end, third-party audit companies perform periodic inspections of the target IT system, log of changes deployed, etc. However, the sheer volume of changes, ever-increasing change complexity, and automation make it challenging to deliver change auditing between inspection events. To tackle this issue, we propose ContAudIT, a blockchain-based approach for continued IT change auditing. In summary, we instrumented a change orchestration framework with a solution for certifying each change deployed in the target system through blockchain. The chain of IT changes in between inspection events is then used to ensure that only certified changes were deployed in the infrastructure.*

Resumo. *Mudanças em TI são essenciais para a operação contínua de organizações modernas, mas o gerenciamento inadequado dessas mudanças pode representar riscos significativos para a continuidade dos negócios. Nesse contexto, auditorias desempenham um papel fundamental em garantir a gerência de mudanças segundo procedimentos homologados. Tradicionalmente, firmas de auditoria realizam inspeções periódicas dos sistemas de TI e registros de mudanças. No entanto, o aumento do volume, complexidade e automação das mudanças torna desafiador assegurar a conformidade entre as inspeções de auditoria. Este artigo propõe o ContAudIT, uma abordagem baseada em blockchain para auditoria contínua de mudanças em TI. A solução integra um framework de orquestração de mudanças com um registro em blockchain que certifica e registra cada mudança implementada. A cadeia de mudanças entre eventos de inspeções permite que auditores verifiquem que apenas mudanças certificadas foram realizadas na infraestrutura.*

1. Introdução

Em um mundo cada vez mais dinâmico, mudanças em sistemas e infraestruturas de TI são essenciais para o sucesso das organizações. No entanto, tais mudanças podem resultar em falhas sistêmicas, vulnerabilidades de segurança e até interrupções de serviço se mal planejadas, causando prejuízos financeiros e danos à reputação [Pandey and Mishra 2014]. Um exemplo recente é o incidente envolvendo a Crowdstrike, onde uma atualização de software realizada pela empresa no contexto de uma mudança de TI mal gerenciada resultou em uma disrupção global de serviços, afetando 8,5 milhões de dispositivos com

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Windows e impactando setores como saúde, aviação, telecomunicações e bancos, com perdas globais estimadas em US$ 5 bilhões [George 2024].

A disciplina de Gerência de Mudanças em TI visa evitar tais cenários problemáticos, propondo boas práticas e processos para garantir a qualidade das mudanças [Axelos 2019, Isaca 2018, Marcel et al. 2024, Mahimkar et al. 2021]. Preocupações com segurança e certificação de procedimentos de mudança também foram incorporadas nesta disciplina, aumentando a confiança das organizações em seus processos de mudanças [Mohan and Othmane 2016]. Nesse contexto, auditorias de TI são cruciais para aferir a conformidade dos processos com metodologias e padrões estabelecidos. Além disso, auditorias independentes são cruciais para acreditar organizações, fomentando parcerias, novos investimentos e a confiança da sociedade [Gantz 2013].

Para fins de auditoria, o estado corrente da infraestrutura de TI de uma organização é analisado regularmente, mais comumente em inspeções anuais [Gantz 2013]. Contudo, o grande volume de mudanças, a crescente complexidade e a maior automação [Han et al. 2022] impõem desafios para certificar mudanças que ocorrem entre inspeções – bem como os estados intermediários da infraestrutura entre essas mudanças. Para ilustrar o problema, considere o exemplo da Figura 1, que mostra mudanças C_i executadas entre inspeções feitas nos instantes D_i. Em cada inspeção, o estado aferido da infraestrutura é dado por S_i. Agora considere as duas inspeções de auditoria realizadas em D_1 e D_2, as quais aferiram (e certificaram) os estados S_1 e S_2. No entanto, logo após D_1 o gerente de TI executou a mudança não certificada C_1, a qual causou um problema na infraestrutura. Até que o problema fosse resolvido com a mudança C_2, a infraestrutura permaneceu em um estado problemático e inconsistente, que não seria homologado pela auditoria. De modo geral, todos os estados intermediários da infraestrutura entre mudanças, no intervalo entre as inspeções S_1 e S_2, serão uma incógnita para os processos de auditoria e certificação.

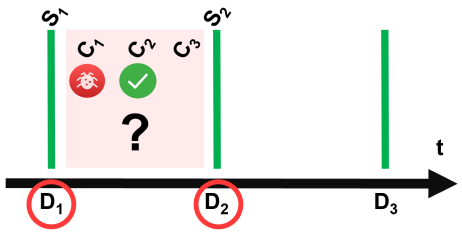


Figura 1. Problema com mudanças não certificadas entre inspeções de auditoria.

O exemplo acima ilustra a dificuldade em garantir conformidade contínua para mudanças realizadas ao longo do tempo [Han et al. 2022, Aditya et al. 2018]. Auditores também enfrentam desafios quando não têm acesso físico à organização ou quando não há evidências históricas disponíveis dos estados dos sistemas e infraestruturas [Zheng et al. 2018]. Nesse contexto, levanta-se a seguinte pergunta de pesquisa neste artigo: "**É possível certificar e aferir todos os estados intermediários entre mudanças que ocorrem entre inspeções de auditoria?**" A tecnologia *blockchain* tem sido recentemente empregada em casos de uso onde auditabilidade e verificação por diferentes partes interessadas em ambientes comerciais ou regulatórios são necessárias. Isso se deve às características nativas da tecnologia, como (i) transparência e rastreabilidade, (ii) segurança dos dados e (iii) descentralização [Elommal and Manita 2022]. Um sistema descentralizado de auditoria contínua poderia fornecer verificação em tempo real de todas as mudanças realizadas nos sistemas e infraestruturas de TI, aprimorando a segurança e a auditabilidade ao eliminar lacunas e garantir um registro constante e imutável dos estados do sistema.

Considerando o exposto, neste artigo propõe-se uma abordagem de auditoria contínua para gerenciamento de mudanças em TI usando *blockchain*. A estratégia envolve a integração de um *framework* de orquestração de mudanças com uma solução baseada em *blockchain* que certifica cada mudança implementada nos sistemas e infraestruturas. Essa abordagem garante que apenas modificações autorizadas tenham sido implementadas desde a última auditoria, fornecendo aos auditores visibilidade da sequência de mudanças certificadas entre inspeções. Como resultado, aumenta-se a precisão e a confiabilidade dos relatórios de auditoria, reforçando a confiança do público no processo de gerenciamento de mudanças da organização. Retomando o exemplo da Figura 1, a abordagem proposta para auditoria contínua visa confirmar que não apenas os estados S_1 e S_2 estão em conformidade, mas também verificar que S_2 pode ser alcançado a partir de S_1 aplicando os efeitos das mudanças executadas C_1, C_2 e C_3. Formalmente, tal pode ser expresso como: $S_{n+1} = S_n + \sum_{i=1}^{k} s(C_i, S_i)$, onde *s($C_i$, S_i)* representa a função que define os efeitos da mudança C_i no estado intermediário S_i, e *k* é o número de mudanças aplicadas. Caso a igualdade não seja satisfeita, haverá indícios de que ocorreu uma mudança não registrada entre inspeções, o que deverá ser destacado pela auditoria.

O presente artigo amplia a pesquisa apresentada em artigo de *poster* [Fraga et al. 2024a] e de *workshop* [Fraga et al. 2024b], onde propusemos uma abordagem preliminar baseada em *blockchain* para auditoria contínua no gerenciamento de mudanças. O estudo atual avança no *framework* anterior, introduzindo um modelo fim a fim de auditoria para infraestruturas em TI. Os resultados da nossa avaliação experimental mostram que a solução proposta reduz efetivamente o esforço manual na coleta de evidências, garantindo a integridade dos dados de auditoria sem impactar negativamente o processo de mudanças, confirmando assim a viabilidade de integrar auditoria contínua com *blockchain*.

O restante deste artigo está organizado da seguinte forma. A Seção 2 revisa os principais trabalhos relacionados. A Seção 3 detalha o problema em investigação e a abordagem proposta de auditoria contínua. A Seção 4 apresenta a avaliação experimental, enquanto a Seção 5 discute as limitações da solução proposta e pesquisas futuras. Finalmente, a Seção 6 conclui com considerações finais.

2. Trabalhos Relacionados

A literatura é rica em trabalhos que destacam a importância de processos de auditoria contínua. Apesar dos esforços em pesquisa, não foi possível identificar uma solução capaz de garantir auditoria automatizada em infraestruturas de TI. A seguir revisamos algumas das pesquisas mais influentes na área. Chan e Vasarhelyi [Chan and Vasarhelyi 2018] mostram que a auditoria contínua introduziu inovações por meio de sete dimensões: (i) auditorias mais frequentes/contínuas, (ii) proatividade, (iii) automação, (iv) evolução do trabalho e dos papéis dos auditores, (v) mudanças na natureza, no momento e no escopo das auditorias, (vi) uso de modelagem de dados e análises para monitoramento e testes e (vii) mudanças no momento e no formato dos relatórios. Além disso, os autores descrevem a auditoria contínua como um processo de quatro etapas: a) automação de procedimentos, b) modelagem de dados e desenvolvimento de benchmarks, c) análise de dados e d) elaboração de relatórios. Segundo os autores, o conceito de auditoria contínua vem sendo cada vez mais adotado por profissionais e acadêmicos, o qual se espera que substitua progressivamente os métodos tradicionais de auditoria. Assim, os autores estabelecem as principais bases teóricas sobre as quais nossa pesquisa está apoiada.

Mahimkar et al. [Mahimkar et al. 2021] propuseram um *framework* para gerenciamento de mudanças que suporta a composição de processos de mudanças, planejamento

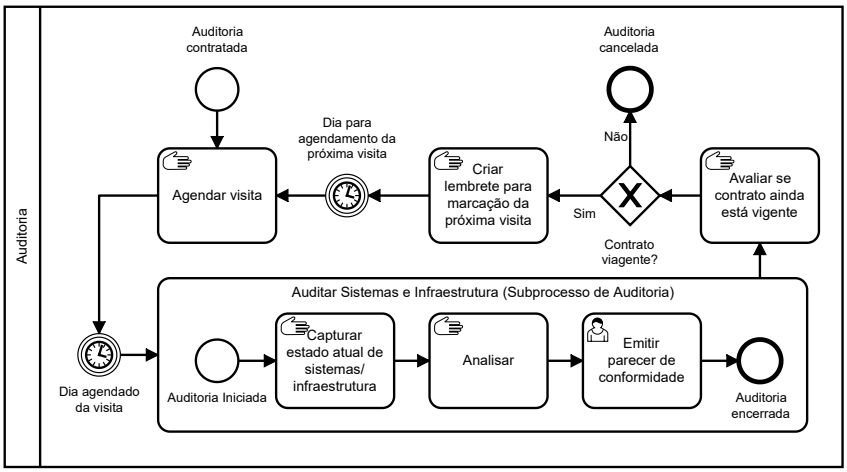


Figura 2. Exemplo de um processo de auditoria realizado por terceiros.

e otimização, verificação de impactos e tradução de mudanças de alto nível em um conjunto de operações que as implementam. No entanto, a auditoria das mudanças executadas pelo *framework* estava fora do escopo do estudo. Rysbekov [Rysbekov 2022] realizou uma pesquisa com engenheiros de DevOps para explorar a viabilidade de conformidade contínua em organizações. O estudo concluiu que há potencial para integrar processos de conformidade aos *pipelines* de desenvolvimento, dado o nível atual de automação nos processos de desenvolvimento e entrega de software. Contudo, o estudo não propõe uma ferramenta para atender às necessidades de auditoria, limitando-se a sugerir aspectos de funcionalidade que poderiam ser explorados em uma solução de conformidade contínua. Algumas dessas sugestões alinham-se à abordagem proposta em nosso estudo, como o uso de um banco de dados *append-only* e o padrão de assinatura *hash* SHA-256. Por fim, Chatziamanetoglou e Rantos [Chatziamanetoglou and Rantos 2023] propuseram um modelo teórico de gerenciamento de configuração utilizando *blockchain*. Embora o estudo compartilhe vários objetivos e funções com o nosso, os autores não desenvolveram uma prova de conceito ou protótipo para avaliar o modelo proposto. Os autores também destacaram a necessidade de pesquisas para explorar a aplicação prática dessa teoria.

Outras investigações com *blockchain* e auditoria incluem De Castro et al. [de Castro et al. 2022], que exploram *blockchain* para auditar operações de processamento de dados em conformidade com a Lei Geral de Proteção de Dados (Brasil) e o *General Data Protection Regulation* (Europa). Vries [de Vries 2022] investiga a detecção de anomalias em auditorias de TI, defendendo a análise de populações inteiras de dados em vez de análise baseada em amostragem. Marques et al. [Marques et al. 2022] propõem uma solução *blockchain* para implementar um log distribuído e auditável usando dados enviados por coletores personalizados autorizados. Hashem et al. [Hashem et al. 2023] examinam como *blockchain* pode melhorar a qualidade da auditoria, destacando melhorias na eficiência de tempo, análise em nível populacional e no estabelecimento de um processo contínuo de auditoria. Apesar dos avanços feitos nesses estudos, permanece uma lacuna significativa em soluções que forneçam auditoria contínua na gerência de mudanças.

3. Contextualização e Proposta

A Figura 2 resume as etapas de uma inspeção de auditoria realizada por terceiros, ilustrando sua interação com a organização auditada [Moeller 2010]. O processo começa com o agendamento de uma visita inicial de inspeção após a contratação da empresa de

auditoria. Na data combinada, a empresa de auditoria inicia o Subprocesso de Auditoria, que envolve a avaliação do estado atual dos sistemas e da infraestrutura, a realização de uma análise e a emissão de um relatório de conformidade. Após a conclusão desse subprocesso, é determinado se o contrato com a organização permanece ativo. Caso positivo, uma visita ou inspeção futura é agendada. No entanto, conforme destacado anteriormente, quaisquer inconformidades entre visitas de auditoria não serão identificadas. Caso contrário, o processo de auditoria não avança.

Como principal contribuição deste artigo, buscamos responder à seguinte questão de pesquisa: *Até que ponto a tecnologia blockchain pode garantir conformidade entre inspeções de auditoria com um sistema descentralizado de auditoria contínua de TI sem perda de desempenho?* Além disso, pretendemos alcançar os seguintes objetivos específicos para aprimorar o processo de auditoria de mudanças para auditoria contínua: (i) reduzir o esforço manual na coleta de evidências para auditorias; (ii) garantir a integridade dos dados coletados; (iii) possibilitar a incorporação de diversos processos de verificação para fins de auditoria dentro dos procedimentos de mudança; e (iv) verificar a integridade dos sistemas e da infraestrutura após cada mudança.

3.1. Coleta Automática de Evidências

Para agilizar o processo de coleta de evidências, propomos que auditorias de TI incorporem uma aplicação, aqui denominada "agente de auditoria", para reunir evidências das execuções de mudanças. Esse agente se integraria de forma transparente aos processos de mudanças da organização com impacto mínimo. Além disso, ele armazenaria as informações coletadas em um repositório, ou seja, a *blockchain*, acessível a partes interessadas relevantes, como auditores externos e a organização auditada.

Definir as informações a serem coletadas dos sistemas e da infraestrutura é responsabilidade da equipe de auditoria e deve ser configurado no agente de auditoria. Um exemplo de modelo de informações que pode ser utilizado é o *Common Information Model* (CIM)[1], mantido pelo *Distributed Management Task Force* (DMTF). Segundo o DMTF, o CIM fornece uma definição padronizada de informações de gerenciamento para sistemas, redes, aplicações e serviços, e permite extensões específicas de fornecedores.

O agente de auditoria deve ser disponibilizado para as organizações auditadas, permitindo que elas o incorporem em seus processos e ferramentas responsáveis pela execução de mudanças. Além disso, o processo de auditoria deve incluir controle sobre as assinaturas das versões do agente de auditoria, garantindo que apenas versões aprovadas possam acessar a *blockchain* para fins de gravação.

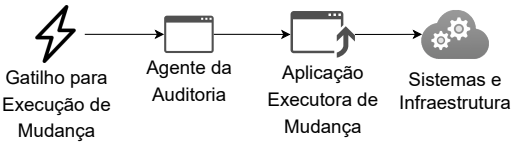


Figura 3. Exemplo de uso do agente, adaptado de [Fraga et al. 2024a].

A Figura 3 ilustra um esquema de execução de mudanças com auditoria, desde o gatilho que inicia a mudança – por ex., a configuração de um cluster em um data center ou a instalação de uma plataforma de software. Esse gatilho ativa o agente de auditoria, responsável por reunir evidências relacionadas à mudança. O agente é pré-configurado ou

[1]https://www.dmtf.org/standards/cim

instalado para interagir com o sistema que orquestra mudanças na organização, como o aplicativo de execução de mudanças (e.g., Jenkins[2], Kubernetes ou o sistema de gerenciamento de pacotes do Ubuntu. Por meio do agente de auditoria, o aplicativo de execução pode ser iniciado. Posteriormente, todas as evidências geradas pelo aplicativo de execução são coletadas pelo agente e transmitidas à *blockchain*. O agente de auditoria deve ser capaz de verificar sua assinatura para garantir que uma validação ocorre para confirmar que a versão em execução do agente é válida e autêntica. Além disso, o agente deve calcular as assinaturas das aplicações de execução de mudanças e quaisquer artefatos necessários para a execução das mudanças. Por exemplo, se o agente estiver configurado para usar o Terraform[3], um arquivo com a extensão ".tf" pode ser considerado um artefato.

3.2. Integridade das Evidências

No projeto, usamos *blockchain* para garantir a integridade das evidências produzidas pela organização para o processo de auditoria. A Figura 4 ilustra a topologia de uma rede *blockchain* usada em nossa abordagem. Cada organização participante (auditoria na cor rosa e organização auditada na cor cinza), identificadas por uma Autoridade Certificadora (CA), implantará quatro contratos inteligentes (S). Além disso, recomendamos a configuração de três nós de ordenação (O) da auditoria para melhorar a disponibilidade e o desempenho, garantindo capacidade de processamento suficiente para os quatro nós pares (P) que lidam com transações dentro da rede (N), provenientes de aplicações externas (A), armazenando os dados no livro-razão de cada par (L) através de um canal (C).

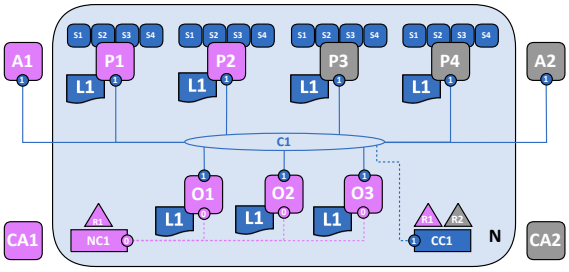


Figura 4. Topologia da rede *blockchain*. Fonte: [Fraga et al. 2024b].

No nosso contexto, a *blockchain* serve como um repositório de dados auditável com total transparência transacional entre os membros da rede. Assim, as partes interessadas em acessar informações tornam-se pares na rede e podem recuperar todos os dados armazenados nesse repositório compartilhado. Além disso, o uso da *blockchain* como repositório de dados em um agente de auditoria confiável garante que todas as evidências geradas por aplicações que executam mudanças nos sistemas e na infraestrutura integrados a ele permaneçam confiáveis. Consequentemente, torna-se imune a adulterações por qualquer parte com intenções maliciosas de comprometer a precisão e a confiabilidade da análise de auditoria. Também é possível implementar um nível mais alto de segurança e restrição para o processo de mudanças, caso seja exigido pela organização. Nesse caso, qualquer mudança só será autorizada se as ferramentas usadas para orquestrá-la tiverem sido previamente comunicadas e aprovadas pela equipe de auditoria. Esse processo pode ser gerenciado utilizando *blockchain*, armazenando informações sobre as ferramentas aprovadas e os artefatos de entrada. Contratos inteligentes podem ser executados para verificar se uma dada ferramenta já foi aprovada, autorizando assim seu uso na mudança.

[2]https://www.jenkins.io
[3]https://www.terraform.io

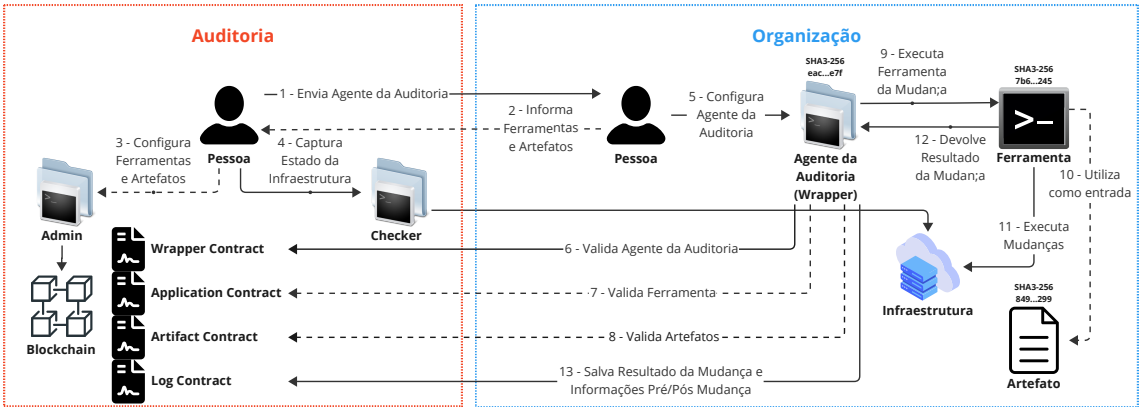


Figura 5. Esquema de exemplo de uso da abordagem proposta, adaptado de [Fraga et al. 2024a].

3.3. Verificação Contínua de Mudanças em TI

Uma vez que o agente de auditoria está configurado e instalado com as aplicações de execução de mudanças (conforme descrito na Seção 3.1), o agente pode avaliar os sistemas e a infraestrutura para diversos fins de conformidade. Por exemplo, ao avaliar mudanças de instalação de software, a auditoria pode examinar as aplicações e bibliotecas instaladas antes e depois da mudança. Comparando esses dois estados, torna-se possível avaliar o impacto da mudança na lista de aplicações instaladas.

As informações de diferenças de estado mencionadas anteriormente, juntamente com informações sobre a aplicação que executa a mudança, artefatos envolvidos e resultados da execução, podem ser armazenadas na *blockchain*. Essa abordagem garante um histórico completo das mudanças realizadas, incluindo os estados intermediários dos sistemas e da infraestrutura de TI antes e depois de cada mudança, conforme necessário para satisfazer o formalismo matemático mencionado na introdução.

3.4. Integridade dos Sistemas e Infraestruturas entre Eventos de Auditoria

A abordagem proposta permite que a auditoria avalie o estado atual dos sistemas e da infraestrutura e compare com os resultados da simulação baseada na sequência de mudanças realizadas durante o período, conforme registrado na *blockchain*. Especificamente, o estado capturado durante o último evento de inspeção pode ser atualizado aplicando as diferenças de estado registradas na *blockchain* em cada mudança realizada. Após a aplicação da última mudança, é necessário apenas verificar se a assinatura do estado simulado coincide com a assinatura do estado atual observado durante a inspeção. Dessa forma, a auditoria introduz um elemento de automação e continuidade, permitindo que os auditores relatem com confiança a integridade e a conformidade da infraestrutura durante o período auditado. A Figura 5 apresenta um diagrama ilustrando a abordagem com um exemplo. No passo #1, um representante da auditoria envia um agente para a organização auditada. No passo #2, se a organização utiliza o modelo de validação restrita de ferramentas e artefatos da auditoria, os dados destas são compartilhados com a equipe da auditoria. No passo #3, a auditoria analisa essas ferramentas e artefatos, aprova-os e os configura na *blockchain* para uso futuro nos processos de mudanças da organização auditada. No passo #4, como esta é a visita inicial de auditoria após a contratação, uma ou mais ferramentas da auditoria realizam uma avaliação geral dos sistemas e da infraestrutura. No passo #5, um representante da organização auditada configura o agente de auditoria dentro das aplicações de execução de mudanças.

A partir do passo #6, o ciclo se repete para cada mudança que a organização implementa usando essas ferramentas. Nesse processo, o agente de auditoria verifica na *blockchain* se a assinatura da versão usada corresponde a alguma assinatura de versão válida do agente. Caso contrário, a mudança não é autorizada e o processo não pode continuar, pois a aplicação pode ser não confiável. Nesses casos, alguém dentro da organização auditada deve reconfigurar uma versão válida do agente. No caso de uma versão confiável do agente, os passos #7 e #8 são executados. Se a assinatura da aplicação que fará a mudança e/ou os artefatos que ela usa não forem válidos e aprovados pela auditoria, a mudança não será autorizada a prosseguir, pois a aplicação será considerada não habilitada.

Após concluir todas as validações, o agente de auditoria avança para o passo #9, invocando a aplicação de execução de mudanças, que pode utilizar artefatos baseados em arquivos (o passo #10). A aplicação então executa as mudanças tanto nos sistemas quanto na infraestrutura, como mostrado no passo #11 do esquema na Figura 5. Após concluir a mudança, a ferramenta de execução retorna seu resultado ao agente de auditoria, que então registra o resultado juntamente com as informações pré e pós-mudança na *blockchain*, conforme ilustrado pelos passos #12 e #13 na Figura 5.

4. Avaliação Experimental

Para avaliar nossa proposta, implementamos um protótipo de prova de conceito desenvolvendo um agente de auditoria e configurando uma rede *blockchain*. Também criamos diversos *workflows* de mudanças no sistema para simular o uso da solução no ambiente de TI de uma organização auditada. O objetivo é avaliar qualitativamente a capacidade da solução de (i) permitir a coleta automática de evidências relacionadas às mudanças, (ii) garantir a integridade dos dados coletados sobre as mudanças, (iii) integrar diversos processos de verificação para fins de auditoria nos procedimentos de mudança da organização e (iv) verificar a integridade dos estados dos sistemas e da infraestrutura entre eventos de auditoria e após cada mudança. Além disso, buscamos avaliar quantitativamente o desempenho da solução, considerando métricas de consumo de recursos computacionais.

4.1. Protótipo de Prova de Conceito

Desenvolvemos cinco projetos distintos para avaliar a abordagem proposta. Nosso código-fonte e os resultados dos experimentos estão disponíveis no GitHub[4]. Também fornecemos os comandos e scripts para executar cada projeto nesse repositório.

Projeto Chaincode. Abrange todos os subprojetos de contratos inteligentes relacionados à *blockchain*, referidos como "*Contract*" na Figura 5, Além disso, todos os projetos foram desenvolvidos para operar em uma rede *blockchain* dentro da infraestrutura do Hyperledger Fabric[5] e as assinaturas de aplicações e arquivos foram calculadas usando o padrão SHA3-256 (2015). São eles:

- ***Wrapper Chaincode*** - mantêm a assinatura válida do agente de auditoria usado pelas organizações durante suas mudanças.
- ***App Chaincode*** - mantêm informações sobre ferramentas usadas pelas organizações auditadas, credenciadas ou aprovadas pela auditoria.
- ***Artifact Chaincode*** - mantêm informações sobre artefatos usados pelas organizações auditadas, que são aprovados pela auditoria.
- ***Log Chaincode*** - mantêm registros das mudanças da organização auditada.

[4]https://github.com/contaudit/contaudit

[5]https://www.hyperledger.org

Projeto Admin. Inclui uma aplicação projetada exclusivamente para uso da auditoria, referida como "Admin" na Figura 5. Ela permite a interação com os contratos inteligentes dos projetos Wrapper Chaincode, App Chaincode e Artifact Chaincode. Auditores podem usar a aplicação para consultar e modificar o *hash* válido do agente distribuído às organizações auditadas, bem como gerenciar a lista de aplicações de execução de mudanças aprovadas e seus artefatos associados.

Projeto Wrapper. Fornece um agente de auditoria, referido como "Audit Agent (Wrapper)" na Figura 5. Ele permite que a empresa de auditoria implemente verificações antes e depois das mudanças. Durante uma mudança, o agente executa procedimentos pré- e pós-mudanças, junto com a mudança em si. O agente consolida todos os dados de execução, incluindo aqueles realizados por ele mesmo e pela aplicação de mudanças da organização, e envia essas informações para a *blockchain* usando o contrato inteligente *Log Chaincode*. Os dados registrados incluem 1) a linha de comando usada para invocar a aplicação de mudanças, 2) detalhes da aplicação de mudança (nome, localização na máquina e *hash*), 3) quaisquer artefatos utilizados pela aplicação de mudança (nome, localização na máquina, *hash* e conteúdo), 4) o resultado dos procedimentos pré-mudança, 5) o resultado da mudança e 6) o resultado dos procedimentos pós-mudança.

Projeto Samples. Reúne ferramentas e artefatos usados para os propósitos de pesquisa neste artigo, referidos como "Ferramenta" e "Artefato" na Figura 5. Como ferramenta, desenvolvemos uma aplicação simples que executa comandos de *shell*, conforme descrito no artefato de entrada. Os artefatos incluem arquivos com a extensão ".workflow", contendo comandos de exemplo para uso potencial em um *workflow* de mudanças. Dois *workflows* criados no contexto deste artigo mostram a instalação de um pacote ou aplicação em uma máquina e a configuração de um cluster Kubernetes em um provedor de nuvem.

Projeto Checker. Envolve o desenvolvimento de uma aplicação projetada especificamente para a auditoria, referida como "Checker" na Figura 5. A aplicação permite auditoria e análise dos sistemas e infraestruturas da organização durante eventos anuais de auditoria, bem como a realização de verificações de conformidade. Essas verificações são realizadas comparando o estado atual dos sistemas e infraestrutura com os estados anteriores registrados em auditorias passadas ou com os dados armazenados na *blockchain*, capturando mudanças feitas ao longo do tempo.

4.2. Testes de Verificação do Protótipo

Para avaliar a abordagem proposta e suas aplicações, definimos um cenário de uso envolvendo uma organização auditada que implementa diversas mudanças em seus sistemas e infraestrutura entre eventos de auditoria. Os experimentos foram conduzidos na nuvem em uma instância *m4.4xlarge* da Amazon Web Services (AWS)[6], provisionada com 16 vCPUs e 64 GiB de RAM, a fim de hospedar a rede *blockchain* e simular múltiplos processos cliente interagindo com o ambiente.

Primeiro, configuramos a rede *blockchain* permissionada Hyperledger Fabric usando a ferramenta Minifabric, seguindo a topologia mostrada na Figura 4. Em seguida, implantamos os contratos inteligentes do projeto Chaincode na rede *blockchain*. Após isso, configuramos a identidade da auditoria nas aplicações dos projetos Admin e Checker, e a identidade da organização na aplicação do projeto Wrapper. A partir daí, usamos a aplicação do projeto Admin para configurar os contratos inteligentes com a assinatura do agente de auditoria, a aplicação de execução de mudanças e os artefatos do *workflow* de mudanças (Samples) no formato SHA3-256. Nesta etapa, podemos começar a medir

[6]https://aws.amazon.com/pt/ec2/instance-types

a visita inicial de auditoria à organização e o processo de implementação de mudanças usando o agente de auditoria. Para isso, usamos a aplicação do projeto Checker para verificar o estado atual dos sistemas e infraestrutura. Neste caso, isso envolve a leitura da lista pacotes e aplicações instaladas na máquina que será o alvo dos testes de mudança. A auditoria pode definir várias ações a serem realizadas durante as inspeções anuais.

Para fins de avaliação, desenvolvemos cinco *scripts* para simular a execução de mudanças na organização. Cada script contempla um *workflow* de mudança diferente e foi executado por meio do agente de auditoria e da aplicação de execução de mudanças da organização. Isso nos permitiu avaliar a eficácia da abordagem na utilização de uma *blockchain* para lidar com mudanças paralelas. Ao executar esses *scripts*, monitoramos a rede usando o Hyperledger Explorer, confirmando que todas as transações das mudanças foram processadas. Além disso, capturamos estatísticas de uso de recursos dos pares da organização (P1 e P2). Os *scripts* foram os seguintes:

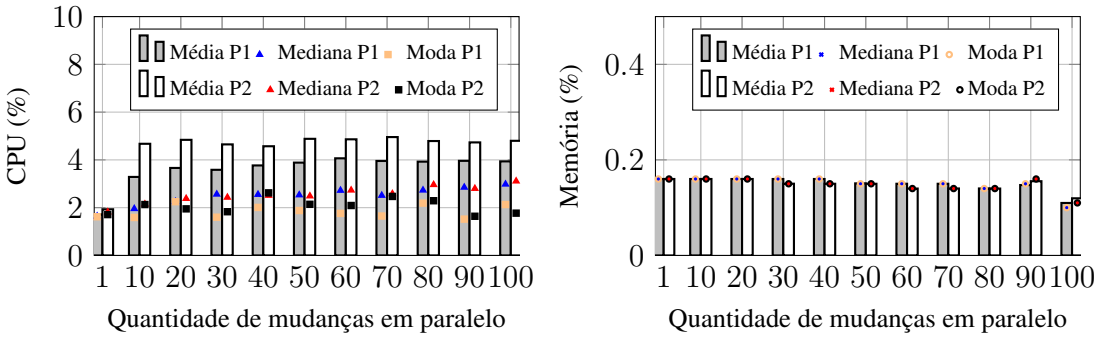
- **Exp. #1**: Avaliamos o desempenho de dois *workflows* (implantação de página Apache e criação de usuário no sistema) sob execuções paralelas variando de 1 a 100 instâncias. Foram coletadas métricas de CPU e memória, e gerados gráficos representando a moda, média e mediana para ambas as métricas.
- **Exp. #2**: Quantificamos os quantis de uso de CPU e memória (Q_0, Q_1, Q_2, Q_3, Q_4) para execuções paralelas de 1 a 100 instâncias, utilizando os mesmos *workflows* e reutilizando os dados de execução do primeiro experimento.
- **Exp. #3**: Medimos os tempos de execução de 100 execuções paralelas, repetidas 30 vezes. A média e o desvio padrão foram calculados, e os resultados apresentados em um gráfico composto por barras e linhas.
- **Exp. #4**: Avaliamos o desempenho de um *workflow* de instalação de pacotes, medindo tempo de execução, consumo de CPU e memória, repetido 30 vezes.
- **Exp. #5**: Executamos um *workflow* mais complexo envolvendo a criação automatizada de um *cluster* Kubernetes (K3S) na Digital Ocean, utilizando Terraform.

Finalmente, durante uma segunda auditoria simulada da organização, usamos a aplicação do projeto Checker para avaliar as assinaturas da primeira e segunda inspeções. Especificamente, verificamos se a assinatura do estado atual da infraestrutura corresponde à assinatura resultante da simulação de todas as mudanças registradas na *blockchain*, aplicadas ao estado obtido durante a inspeção anterior. Esse processo garante que as mudanças estão em conformidade com o formalismo matemático descrito na introdução.

4.3. Análise Quantitativa do Desempenho

A Figura 6 mostra as porcentagens de uso de CPU e memória (média, mediana e moda) no Exp. #1. Para o P1, o uso médio de CPU estabiliza-se em 3,9% à medida que o número de mudanças paralelas aumenta, enquanto a mediana permanece inferior, em torno de 2,5%. Em contraste, o P2 apresenta um uso médio de CPU mais elevado, atingindo um pico de cerca de 4,8%. O uso de memória mostrou-se ligeiramente menor entre os pares da organização à medida que o número de mudanças paralelas aumenta. Para o P1, o uso médio de memória começa em 0,16% e gradualmente declina para 0,11% após 100 mudanças paralelas, com a mediana e a moda apresentando valores semelhantes ao longo do tempo. O P2 segue uma tendência comparável, com a média diminuindo de 0,16% para 0,12%, e uma mediana ligeiramente superior até 90 mudanças paralelas.

A Tabela 1 apresenta os dados consolidados de uso de recursos no Exp. #2. Note que os quartis de CPU em geral alinharam-se com as médias, exceto para P(100), que representa o valor máximo medido. Nesse caso, foram observados valores nominais de até 20% de uso de CPU. O uso de memória também manteve regularidade similar.



(a) CPU (%) por *peer* no *workflow* #1. **(b) Memória (%) por *peer* no *workflow* #1.**

Figura 6. Resultados para execuções paralelas (de 1 a 100) do experimento #1.

Tabela 1. Uso de CPU por *peer* no primeiro *workflow* de mudança do exp. #2.

Peers da Organização	Mudanças em paralelo	Quantis				
		Q_0 (mín)	Q_1	Q_2 (mediana)	Q_3	Q_4 (máx)
Peer 1	25	1.43	1.9	2.68	5.1525	14.44
	50	1.22	1.89	2.53	4.8075	18.13
	75	0.87	1.97	2.72	4.65	20.1
	100	0.06	1.99	2.98	4.61	32.03
Peer 2	25	1.27	2.015	2.7	7.3325	14.49
	50	1.19	2.1175	2.485	7.74	17.06
	75	0.88	2.25	3.11	7.45	19.07
	100	0.06	1.95	3.11	6.87	23.78

A Figura 7 apresenta os tempos medidos para o Exp. #3. Observe que, para o mesmo fluxo de mudança e número de execuções paralelas, o tempo total para executar todas as mudanças permaneceu quase constante em 30 repetições desse cenário. Isso indica que a abordagem/solução proposta não causa degradação de desempenho na execução de mudanças ao longo do tempo, desde que a configuração de recursos e a topologia da rede permaneçam inalteradas. O tempo médio por reepetiçao varia de 137,84 a 143,03 segundos, com pequenas variações.

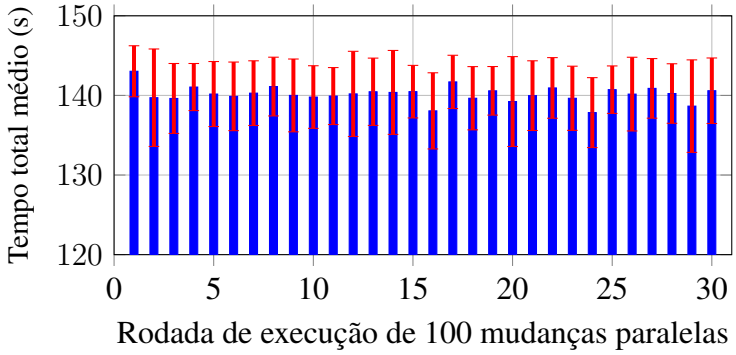
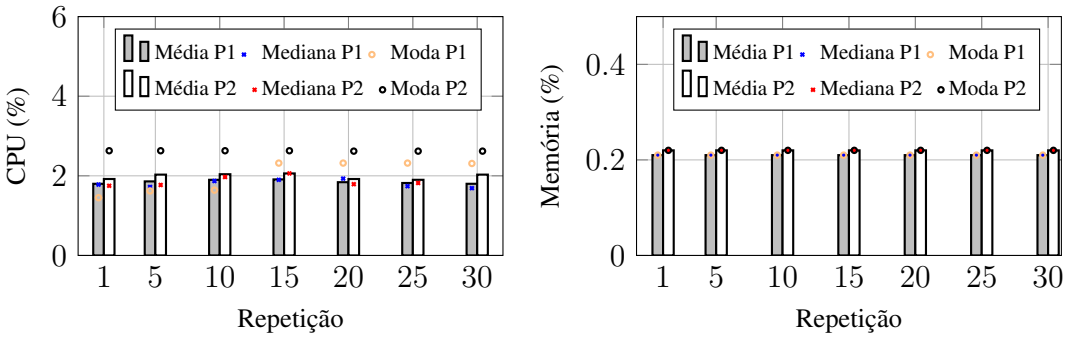


Figura 7. Tempo total médio e desvio padrão por rodada de execução no exp. #3.

Por fim, a Figura 8 apresenta os resultados para o Exp. #4. Nele, o mesmo fluxo de mudança foi repetido 30 vezes, mas sem execuções paralelas. Similar ao Exp. #3, o tempo total de execução permaneceu estável em todas as rodadas. Finalmente, no Exp.



(a) CPU (%) por *peer*. **(b) Memória (%) por *peer*.**

Figura 8. Medições de uso de CPU e de memória para o Exp. #4.

#5, um fluxo de mudança mais complexo foi executado uma única vez, também sem paralelismo. Em todos os casos, as transações foram confirmadas com sucesso na *blockchain* após a conclusão das etapas de execução. Em resumo, do ponto de vista quantitativo, considera-se marginal a sobrecarga de execução do agente de auditoria contínua e o armazenamento de dados na *blockchain*. O principal custo computacional nos cenários avaliados é atribuído ao espaço em disco necessário para armazenar a *blockchain*, que é mantida pela empresa de auditoria.

4.4. Análise Qualitativa dos Objetivos

Ao analisar os resultados da execução do agente de auditoria, permite-se inferir uma redução significativa no esforço manual exigido tanto da organização quanto da equipe de auditoria para obter evidências das mudanças realizadas. Do ponto de vista da integridade dos dados, como o agente de auditoria é instalado diretamente na máquina que executa a aplicação de mudança, verifica-se que as informações recebidas pelo agente são completas. Isso ocorre porque não há intermediários entre o agente de auditoria e a aplicação de execução de mudanças. Assim, as informações coletadas refletem diretamente a saída relatada pela aplicação que realizou as mudanças. Além disso, o agente de auditoria está diretamente conectado à *blockchain* para armazenar informações de forma segura, invocando o contrato inteligente *Log Chaincode*. Isso garante que os dados permaneçam inalterados desde o agente até a rede *blockchain*, pois o protocolo empregado pela aplicação para se comunicar com a *blockchain* é o gRPC (Google RPC) sobre HTTPS. Adicionalmente, os recursos inerentes à integridade de dados da *blockchain* – como a sequencialização e assinatura de transações em blocos distribuídos entre os participantes da rede (referidos como nós pares ou *peers*) – reforçam ainda mais a integridade dos dados das mudanças das organizações auditadas.

Confirmamos que, uma vez que o agente de auditoria está integrado aos processos de execução de mudanças, ele permite que as auditorias realizem verificações nos sistemas e infraestrutura para cada mudança, tanto nas etapas pré- quanto pós-mudança. A auditoria também pode aproveitar os dados dessas duas etapas para identificar diferenças nos sistemas e infraestrutura. Portanto, a auditoria não apenas adquire a capacidade de incorporar diversos processos de verificação de conformidade às mudanças, mas também avança suas práticas para um nível de auditoria contínua. Isso é alcançado porque essas verificações são realizadas independentemente da presença ou do agendamento formal de uma inspeção de auditoria. Por fim, foi confirmado que o estado dos sistemas e infraestrutura observado durante as inspeções de auditoria alinhou-se com os achados da visita anterior, levando em conta as mudanças subsequentes de TI realizadas ao longo do

período. Como resultado, a auditoria agora pode emitir pareceres de conformidade para todo o período entre as inspeções com um grau maior de confiabilidade. Isso é verdadeiro, pois a auditoria não avalia mais apenas amostras dos estados dos sistemas e infraestrutura em pontos específicos, mas revisa todo o histórico de mudanças no período avaliado.

5. Discussão sobre Limitações e Trabalhos Futuros

Uma premissa fundamental da abordagem proposta é a colaboração da organização auditada. Isso é essencial porque o agente de auditoria deve ser instalado, configurado e utilizado nos processos e ferramentas de mudanças da organização. Considerando os benefícios mútuos para ambas as partes ao adotar essa abordagem, e levando em conta que as organizações geralmente estão sujeitas a algum nível de auditoria durante relações comerciais, é razoável supor que existem as condições necessárias para viabilizar essa colaboração. Além disso, a abordagem proposta oferece mecanismos para detectar não conformidades por parte das empresas auditadas em relação ao uso do agente de auditoria.

Quanto aos experimentos realizados, as aplicações de prova de conceito foram desenvolvidas e testadas exclusivamente em Java sobre GNU/Linux. No entanto, espera-se que o conceito funcione efetivamente em outros sistemas compatíveis com Java. Além disso, empresas de auditoria podem desenvolver aplicações semelhantes em outras linguagens ou para diferentes sistemas, eliminando assim essa limitação. Por fim, há oportunidades para aumentar os níveis de segurança da abordagem proposta via integração de *Hardware Security Modules* (HSM) [Mavrovouniotis and Ganley 2014] às máquinas dos nós da *blockchain*, garantindo que a chave privada da organização permaneça inacessível no sistema de arquivos.

6. Considerações Finais

Mudanças em sistemas e infraestruturas de Tecnologia da Informação (TI) são uma necessidade crucial e contínua para organizações que dependem fortemente de TI. Quando se trata de grandes organizações, processos de auditoria contínua são mandatórios. No entanto, as soluções existentes não consideram a possibilidade de auditar continuamente mudanças em TI, garantindo que a infraestrutura permaneceu em um estado consistente entre inspeções. Para avançar nessa lacuna, este artigo apresentou uma abordagem de auditoria contínua para o gerenciamento de mudanças utilizando a tecnologia *blockchain*. Os resultados alcançados, embora promissores, indicam direções de pesquisa importantes a serem perseguidas pela comunidade de pesquisa. Por exemplo, nossos experimentos indicaram a possibilidade de evoluir o Checker visando reduzir ou mesmo eliminar a necessidade de inspeções frequentes de auditoria. Além disso, há potencial para expandir o Wrapper, permitindo que as organizações utilizem essa ferramenta em conjunto com as auditorias para realizar verificações adaptadas aos seus interesses, abrangendo seus sistemas e infraestrutura, incluindo segurança da informação.

Referências

Aditya, B. R., Ferdiana, R., and Santosa, P. I. (2018). Toward modern IT audit- current issues and literature review. *Proceedings of the 4th ICST 2018*, 1:1–6.

Axelos (2019). *ITIL Foundation*. The Stationery Office, 4th edition.

Chan, D. Y. and Vasarhelyi, M. A. (2018). Innovation and practice of continuous auditing. *Continuous Auditing: Theory and Application*, pages 271–283.

Chatziamanetoglou, D. and Rantos, K. (2023). Blockchain-based security configuration management for ict systems. *Electronics*, 12(8):1879.

de Castro, M., Pereira, M., and de Castro, M. (2022). Uma arquitetura baseada em blockchain para auditoria de conformidade com regulamentos de proteção de dados. In *Anais do SBSeg 2022*, pages 390–395, Porto Alegre, RS, Brasil. SBC.

de Vries, T. (2022). Anomaly detection in IT audit: The possibilities and potential in the domain of IT audit. Master's thesis, University of Turku, Amsterdam, Netherlands.

Elommal, N. and Manita, R. (2022). How blockchain innovation could affect the audit profession: A qualitative study. *Journal of Innov. Econom. & Mgmt*, 37(1):37–63.

Fraga, C., Abelém, A., Borges, V., Pinheiro, B., and Cordeiro, W. (2024a). A blockchain-based approach for continuous auditing in IT change management. In *IEEE/IFIP NOMS 2024 Poster Session*, pages 1–4.

Fraga, C., Abelém, A., Borges, V., Pinheiro, B., and Cordeiro, W. (2024b). Uma abordagem de auditoria contínua com blockchain para gerenciamento de mudanças em TI. In *VII Workshop em Blockchain*, pages 83–96, Porto Alegre, RS, Brasil. SBC.

Gantz, S. D. (2013). *The Basics of IT Audit*. Syngress.

George, D. A. S. (2024). When trust fails: Examining systemic risk in the digital economy from the 2024 crowdstrike outage. *PUMRJ*, 1(2):134–152.

Han, H., Fei, S., Yan, Z., and Zhou, X. (2022). A survey on blockchain-based integrity auditing for cloud data. *Digital Communications and Networks*.

Hashem, R. E. E. D. R., Mubarak, A.-R. I., and Abu-Musa, A. A. E.-S. (2023). The impact of blockchain technology on audit process quality: An empirical study on the banking sector. *International Journal of Auditing and Accounting Studies*, 5(1):87–118.

Isaca (2018). *COBIT 2019 Framework: Introduction and Methodology*. Isaca.

Mahimkar, A., De Andrade, C., Sinha, R., and Rana, G. (2021). A composition framework for change management. In *ACM SIGCOMM 2021 Conf.*, pages 788–806.

Marcel, M., Kristiani, E., and Mudita, D. S. (2024). Enhancing IT change management through communities of practice and social learning: A case study at a university. *Journal of Information Systems and Informatics*, 6(2):1300–1316.

Marques, M., Jr., M. S., and Miers, C. (2022). Event2ledger: Container traceability using docker swarm and consortium hyperledger blockchain. In *Anais do SBSeg 2022*, pages 103–110, Porto Alegre, RS, Brasil. SBC.

Mavrovouniotis, S. and Ganley, M. (2014). *Hardware Security Modules*. Springer.

Moeller, R. R. (2010). *IT Audit, Control, and Security*. Wiley.

Mohan, V. and Othmane, L. B. (2016). SecDevOps: is it a marketing buzzword? mapping research on security in devops. In *ARES 2016*, pages 542–547. IEEE.

Pandey, A. and Mishra, S. (2014). Understanding IT change management challenges at a financial firm. *2014 Information Systems Educators Conf. (ISECON)*, pages 1–10.

Rysbekov, A. (2022). Continuous compliance: Devops approach to compliance and change management. Master's thesis, University of Oslo, Oslo, Norway.

Zheng, Z., Xie, S., Dai, H. N., Chen, X., and Wang, H. (2018). Blockchain challenges and opportunities: A survey. *Intl Journal of Web and Grid Services*, 14(4):352–375.