RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.

Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

Governança orientada por evidências e governance drift em infraestruturas nativas de nuvem

Evidence-oriented governance and governance drift in cloud-native infrastructures

Gobernanza orientada por evidencias y governance drift en infraestructuras nativas de la nube

Obede Bessa Rocha da Silva
Especialista independente em arquitetura de soluções em nuvem, DevOps e governança de infraestrutura digital

RESUMO
Este artigo propõe um modelo conceitual de governança orientada por evidências para analisar a divergência entre intenção institucional aprovada e estado operacional em infraestruturas nativas de nuvem. Realizou-se revisão narrativa estruturada e não exaustiva de normas, especificações, pesquisas revisadas por pares e relatórios técnicos, organizada em cinco categorias: representação da intenção, automação da mudança, identidade e proveniência, observação do estado e tratamento do desvio. A síntese delimita governance drift em relação a desvios de configuração e violações isoladas de política, descreve um ciclo de seis etapas que liga intenção, estado declarado, entrega, estado observado, classificação de materialidade e resposta, e apresenta taxonomia de desvios e modelo de maturidade em cinco níveis. Argumenta-se que governança persistente depende de evidências consultáveis, identidades estáveis, comparação semântica e respostas proporcionais ao risco. O trabalho é conceitual: não valida escala, limiares universais nem eficácia organizacional. Sua contribuição consiste em tornar explícitas as relações e os critérios necessários para futura operacionalização e avaliação empírica do governance drift.
Palavras-chave: governança digital; governance drift; cloud-native; DevSecOps; proveniência; gestão de configuração.

ABSTRACT
This article proposes an evidence-oriented governance model for analyzing divergence between approved institutional intent and operational state in cloud-native infrastructures. A structured, non-exhaustive narrative review of standards, specifications, peer-reviewed research, and technical reports was organized into five categories: intent representation, change automation, identity and provenance, state observation, and deviation treatment. The synthesis distinguishes governance drift from configuration drift and isolated policy violations, describes a six-stage cycle linking intent, declared state, delivery, observed state, materiality classification, and response, and presents a deviation taxonomy and a five-level maturity model. Persistent governance is argued to depend on queryable evidence, stable identities, semantic comparison, and risk-proportionate responses. The work is conceptual: it validates neither a scale, universal thresholds, nor organizational effectiveness. Its contribution is to make explicit the relationships and criteria required for future operationalization and empirical evaluation of governance drift.
Keywords: digital governance; governance drift; cloud-native; DevSecOps; provenance; configuration management.

1 INTRODUÇÃO

Arquiteturas nativas de nuvem substituem conjuntos relativamente estáveis de servidores por ecossistemas compostos por microsserviços, contêineres, controladores, pipelines de integração e entrega contínuas e interfaces declarativas. Essa mudança aumenta velocidade e elasticidade, mas também amplia os pontos em que uma decisão autorizada pode deixar de corresponder ao estado

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

efetivamente executado. O desafio de governança já não é apenas aprovar uma mudança: é conseguir reconstruir quem a autorizou, quais artefatos foram produzidos, como chegaram à produção, qual estado resultou e se a divergência permaneceu dentro do risco aceito.

A gestão de configuração tradicional fornece princípios de baseline, controle de mudança e monitoramento, enquanto infraestrutura como código e GitOps tornam parte da intenção versionável e reconciliável (NIST, 2019; RAHMAN; MAHDAVI-HEZAVEH; WILLIAMS, 2019; BEETZ; HARRER, 2022). Essas capacidades, contudo, não asseguram que a configuração declarada tenha autoridade institucional, que a cadeia de entrega permaneça íntegra ou que o estado observado conserve vínculo verificável com aprovações e exceções. A literatura sobre proveniência e segurança da cadeia de software cobre parte dessa lacuna, mas não substitui a decisão de governança sobre materialidade, legitimidade e resposta (W3C, 2013; TORRES-ARIAS et al., 2019; SLSA COMMUNITY, 2025).

Neste artigo, governance drift designa a divergência material entre intenção institucional autorizada e realidade operacional observável. A intenção pode estar expressa em políticas, decisões arquiteturais, configurações declaradas, requisitos regulatórios e exceções aprovadas. A realidade operacional inclui recursos em execução, identidades, relações, permissões e evidências produzidas durante entrega e operação. A divergência só adquire significado de governança quando é interpretada em relação a autoridade, risco, alcance, duração e responsabilidade; diferença técnica isolada não basta.

O estudo responde a três perguntas: RQ1 — como distinguir governance drift de desvios técnicos adjacentes? RQ2 — quais evidências ligam intenção aprovada, entrega e estado observado? RQ3 — como classificar e tratar divergências sem confundir automação com controle absoluto? As contribuições são: definição delimitada do construto; ciclo analítico de seis etapas; taxonomia de desvios e respostas; modelo de maturidade orientado por capacidades verificáveis; e agenda de validação empírica.

A relevância prática decorre da escala alcançada pelo substrato cloud-native. Na pesquisa anual da CNCF de 2023, 66% dos consumidores potenciais ou atuais respondentes declararam usar Kubernetes em produção e 18% afirmaram avaliá-lo (CNCF, 2024). Esses percentuais não demonstram maturidade de governança nem representam universalmente todas as organizações, mas indicam que decisões automatizadas e estados efêmeros se tornaram parte ordinária da infraestrutura digital.

2 METODOLOGIA

Foi realizada revisão narrativa estruturada e não exaustiva, adequada ao objetivo de integrar conceitos

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

provenientes de engenharia de software, segurança, governança e operação, sem estimar efeitos agregados. A qualidade da revisão foi orientada pelos seis domínios da SANRA: relevância, objetivo, descrição da busca, referenciação, raciocínio científico e apresentação dos dados pertinentes (BAETHGE; GOLDBECK-WOOD; MERTENS, 2019). A SANRA foi empregada como instrumento crítico, não como protocolo de revisão sistemática.

A busca dirigida foi concluída em 25 de agosto de 2026 em IEEE Xplore, ACM Digital Library, ScienceDirect e USENIX, além dos repositórios oficiais do NIST, ISO, W3C, CNCF, DORA e SLSA. Foram combinados, em português e inglês, os termos governance drift, configuration drift, policy as code, infrastructure as code, GitOps, cloud-native governance, continuous monitoring, software provenance, SLSA, DevSecOps e software supply chain. A busca cobriu publicações até a data de corte, sem limite inferior rígido, porque normas fundacionais de proveniência e monitoramento permanecem tecnicamente vigentes.

Foram incluídos documentos com autoria e origem rastreáveis que contribuíssem diretamente para pelo menos uma das cinco categorias analíticas: representação da intenção; automação da mudança; identidade e proveniência; observação do estado; e tratamento do desvio. Priorizaram-se normas e especificações primárias, pesquisas revisadas por pares e relatórios com método descrito. Foram excluídos materiais promocionais, duplicatas, textos sem autoria identificável e fontes que apenas mencionavam ferramentas sem discutir mecanismos, evidências ou riscos. O corpus final reuniu 21 documentos.

A síntese foi realizada por comparação conceitual. Para cada fonte, registraram-se objeto de controle, autoridade presumida, artefato observado, garantia oferecida, risco residual e resposta prevista. Em seguida, os mecanismos foram organizados no ciclo proposto e submetidos a três testes de coerência: distinção entre disponibilidade de evidência e correção semântica; separação entre autorização e configuração; e preservação da responsabilidade humana. Não houve avaliação de organização, experimento, entrevista ou mensuração de incidentes.

Como limitações metodológicas, a busca não pretende cobertura bibliométrica exaustiva, não realizou dupla triagem independente e combina literatura científica com normas e evidência técnica institucional. Consequentemente, a síntese sustenta plausibilidade conceitual e rastreabilidade das proposições, mas não prevalência, causalidade, limiares universais ou eficácia do modelo em produção.

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.
Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

3 FUNDAMENTAÇÃO E DELIMITAÇÃO CONCEITUAL

3.1 Governança como capacidade sociotécnica

Governança digital define direitos decisórios, limites de autonomia, critérios de aceitação de risco e formas de prestação de contas. Em ambientes nativos de nuvem, esses elementos atravessam repositórios, catálogos de serviços, pipelines, registries, clusters e plataformas de observabilidade. O NIST SP 800-53 Rev. 5 e a ISO/IEC 27001:2022 situam controles dentro de processos organizacionais de gestão de risco, evitando reduzi-los a uma lista aplicada depois da implantação (NIST, 2020; ISO, 2022a).

A natureza sociotécnica decorre da interdependência entre código, infraestrutura e coordenação humana. Uma política tecnicamente correta pode falhar sem proprietário, caminho de remediação ou exceção documentada; um processo bem desenhado pode ser anulado por credenciais excessivas, mudanças manuais e fontes de verdade concorrentes. Estudos de DevOps associam desempenho sustentável à combinação de capacidades técnicas, culturais e de aprendizagem, e pesquisas de infraestrutura como código mostram que adoção não elimina desafios de teste, manutenção e suporte (FORSGREN; HUMBLE; KIM, 2018; GUERRIERO et al., 2019).

Controladores declarativos tornam explícito um estado desejado e executam ciclos de reconciliação, mas sistemas distribuídos admitem divergências transitórias, falhas parciais e intervenções emergenciais. A experiência acumulada em Borg, Omega e Kubernetes demonstra a centralidade de mecanismos de controle contínuo e convergência, sem implicar que toda diferença de estado seja uma violação de governança (BURNS et al., 2016). A análise exige autoridade, contexto e materialidade.

3.2 Governança drift e conceitos adjacentes

O Quadro 1 delimita o conceito proposto. Configuration drift descreve diferença entre baseline ou configuração desejada e configuração observada. Violação de policy as code ocorre quando uma regra executável não é satisfeita. Lacuna de proveniência aparece quando não é possível demonstrar origem, processo ou identidade. Governance drift pode incluir qualquer uma dessas condições, mas requer um elemento adicional: divergência relevante entre realidade operacional e intenção institucional autorizada.

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.

Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

Quadro 1 – Delimitação entre governance drift e conceitos adjacentes

Conceito	Objeto comparado	Questão central	Resposta típica
Configuration drift	Baseline/configuração desejada × configuração observada	O estado técnico divergiu?	Reconciliar ou atualizar a baseline
Violação de política	Estado ou artefato × regra executável	Uma condição codificada foi descumprida?	Bloquear, alertar ou abrir exceção
Lacuna de proveniência	Artefato/estado × origem, processo e identidade	A cadeia pode ser verificada?	Preservar, correlacionar ou reconstruir evidências
Governance drift	Intenção autorizada × realidade operacional	A divergência altera risco, legitimidade ou prestação de contas?	Classificar, conter, reconciliar, aceitar ou investigar

Fonte: elaborado pelo autor com base em NIST (2019, 2022a, 2024), W3C (2013), Rahman, Mahdavi-Hezaveh e Williams (2019), Beetz e Harrer (2022) e SLSA Community (2025).

A distinção impede dois erros. O primeiro é classificar qualquer mudança emergencial como falha, mesmo quando autorizada, registrada e temporária. O segundo é presumir conformidade porque uma configuração coincide com o repositório, embora o repositório não represente a decisão vigente ou a cadeia de entrega não possa ser verificada. Governance drift é, portanto, uma relação entre estados e autoridade, não um atributo isolado de um arquivo.

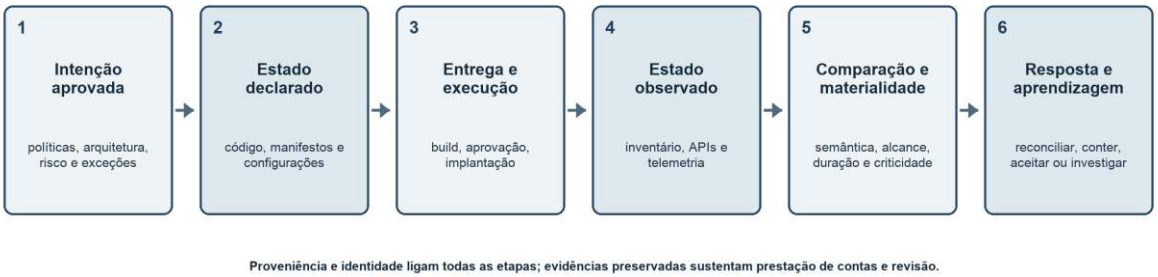
4 MODELO DE GOVERNANÇA ORIENTADA POR EVIDÊNCIAS

O modelo organiza governança persistente em seis etapas encadeadas. Cada etapa produz ou consome evidências, mas nenhuma fonte isolada prova correção do conjunto. A verificabilidade depende de identidade estável, semântica temporal e capacidade de relacionar decisões, artefatos, atividades e estados.

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.
Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

Figura 1 – Ciclo de governança orientada por evidências



Fonte: elaborado pelo autor.

4.1 Intenção aprovada e estado declarado

A intenção aprovada abrange políticas, decisões arquiteturais, critérios de risco, requisitos regulatórios e exceções. A primeira exigência é declarar qual artefato possui autoridade para cada decisão e por quanto tempo. A ISO/IEC 27001:2022 exige que o sistema de gestão de segurança seja adequado ao contexto e aos riscos da organização, enquanto o NIST SSDF recomenda integrar práticas de desenvolvimento seguro ao ciclo de vida (ISO, 2022a; NIST, 2022b).

O estado declarado traduz parte dessa intenção em código de infraestrutura, manifestos, catálogos e regras executáveis. Infraestrutura como código melhora versionamento e repetibilidade, mas estudos apontam limitações de teste, manutenção e apoio ferramental (RAHMAN; MAHDAVI-HEZAVEH; WILLIAMS, 2019; GUERRIERO et al., 2019). GitOps acrescenta reconciliação baseada em repositórios, sem resolver sozinho autoridade, conflito entre fontes ou legitimidade de exceções (BEETZ; HARRER, 2022).

4.2 Entrega, execução e estado observado

Entre declaração e execução existem builds, verificações, assinaturas, aprovações, mutações de admissão e ações de controladores. Falha parcial, dependência indisponível ou intervenção manual pode produzir resultado diferente do planejado. Por isso, comparar apenas repositório e configuração gerada é insuficiente: o estado observado deve incluir inventário, APIs dos planos de controle, telemetria, identidades e relações do runtime.

Policy as code permite antecipar controles, mas regras codificadas continuam sendo software: podem conter erros, ficar desatualizadas e produzir efeitos sistêmicos. O NIST SP 800-204C descreve policy as code e observability as code como componentes do DevSecOps cloud-native, e o NIST SP 800-204D integra controles da cadeia de software aos pipelines (NIST, 2022a, 2024). A governança deve

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

testar as regras, versioná-las, monitorar impacto e preservar procedimentos de exceção.

4.3 Proveniência como ligação verificável

Proveniência descreve entidades, atividades e agentes envolvidos na produção de um artefato ou estado. O modelo PROV do W3C fornece vocabulário interoperável para representar essas relações (W3C, 2013). Na cadeia de software, in-toto, SLSA e Sigstore demonstram mecanismos complementares de declaração de etapas, atestação, assinatura vinculada a identidades e transparência (TORRES-ARIAS et al., 2019; NEWMAN; MEYERS; TORRES-ARIAS, 2022; SLSA COMMUNITY, 2025).

Metadados, por si só, não garantem confiabilidade. Se o mesmo ator puder alterar artefato e registro sem detecção, a evidência terá independência limitada. Controles mais fortes usam separação de funções, identidades de curta duração, builders controlados, assinaturas verificáveis e armazenamento com retenção adequada. A cadeia também precisa ser consultável: identificadores imutáveis, como digests, reduzem correlações manuais e permitem localizar artefatos afetados por vulnerabilidade ou incidente.

A coleta deve obedecer minimização e finalidade. Evidências não devem registrar segredos ou dados pessoais desnecessários. Governança orientada por evidências não equivale a vigilância integral; busca o conjunto suficiente para sustentar decisão, responsabilização e reconstrução, com acesso e retenção proporcionais ao risco.

5 DETECÇÃO, CLASSIFICAÇÃO E RESPOSTA

Detectar drift exige função de comparação e modelo de materialidade. Diferença sintática pode ser irrelevante, enquanto pequena alteração semântica — por exemplo, ampliar uma política de rede ou trocar a identidade de execução — pode modificar substancialmente o risco. Comparadores devem normalizar representações, conhecer valores padrão, considerar relações entre recursos e registrar o instante de validade de cada observação.

O monitoramento contínuo do NIST SP 800-137 fornece princípios aplicáveis: visibilidade sobre ativos, ameaças, vulnerabilidades e efetividade dos controles (NIST, 2011). Em ambientes efêmeros, a frequência da coleta deve acompanhar a velocidade de mudança. Eventos de controle, snapshots assinados e consultas regulares formam evidências complementares; verificações periódicas lentas podem não capturar recursos que existiram por minutos.

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.

Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

Quadro 2 – Taxonomia operacional de divergências

Classe	Condição	Tratamento padrão	Evidência mínima
Autorizada e temporária	Mudança aprovada, com prazo e proprietário	Monitorar e reconciliar no vencimento	Aprovação, escopo, validade e estado
Operacional benigna	Variação esperada sem alteração material de risco	Registrar ou normalizar comparação	Regra de tolerância e observação
Drift de controle	Controle ausente, desatualizado ou contornado	Corrigir controle e avaliar cobertura	Versão da política, resultado e exceção
Drift de segurança	Exposição material de confidencialidade, integridade ou disponibilidade	Conter, preservar evidência e investigar	Linha temporal, identidade, alcance e artefatos
Indeterminado	Evidência insuficiente ou conflitante	Abster-se de correção destrutiva e coletar contexto	Lacunas e grau de confiança declarados

Fonte: elaborado pelo autor.

A classe indeterminada é necessária para evitar certeza artificial. Correção automática baseada em evidência incompleta pode apagar sinais úteis, interromper serviços ou ampliar uma interpretação equivocada. Para riscos baixos e regras bem testadas, a reconciliação automática é plausível; para riscos altos ou baixa confiança, preservação do contexto, contenção e revisão humana devem anteceder alterações irreversíveis.

Indicadores úteis incluem tempo entre desvio e detecção, tempo até reconciliação, proporção de mudanças fora do fluxo aprovado, cobertura de recursos por política, taxa de exceções vencidas e completude das ligações de proveniência. Esses indicadores devem apoiar aprendizagem, não ranquear indivíduos. Metas isoladas incentivam otimização local; a leitura precisa considerar incidentes, carga de trabalho, experiência dos desenvolvedores e resultados de confiabilidade.

6 CONTROLES INCORPORADOS AO CICLO DEVSECOPS

Inserir governança em pipelines antecipa evidências e reduz o custo de correção. Controles podem verificar dependências, segredos, código de infraestrutura, imagens, assinaturas e políticas antes da promoção, mantendo resultados e aprovações. Continuous delivery já trata build, teste e implantação como fluxo automatizado; o diferencial da governança orientada por evidências é conservar a ligação

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.
Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

entre cada resultado, a decisão que o autorizou e o estado que efetivamente surgiu (HUMBLE; FARLEY, 2010).

Shift left não transfere toda responsabilidade aos desenvolvedores. Equipes de segurança e plataforma devem fornecer regras compreensíveis, ferramentas confiáveis e caminhos de exceção. Controles com mensagens opacas e falsos positivos elevados incentivam bypass. O relatório DORA de 2024 associa plataformas internas a ganhos de produtividade e desempenho organizacional, mas também registra reduções de throughput e estabilidade em certos contextos; experiência do usuário, independência e adoção como produto são condições de desenho, não detalhes cosméticos (GOOGLE CLOUD; DORA, 2024).

Controles preventivos precisam ser combinados a mecanismos detectivos e responsivos. Vulnerabilidades emergem após a implantação, dependências mudam e identidades podem ser comprometidas. Políticas de runtime, observabilidade e resposta cobrem riscos residuais. A segregação de funções deve ser proporcional: serviços críticos podem exigir revisão independente, enquanto mudanças de baixo risco podem seguir automação com limites claros.

O princípio central não é multiplicar aprovações, mas impedir que uma única identidade altere intenção, artefato, evidência e produção sem possibilidade de detecção. Assinaturas e logs de transparência reduzem esse risco, mas sua garantia depende de modelo de ameaça, verificação e proteção das identidades (NEWMAN; MEYERS; TORRES-ARIAS, 2022).

7 MODELO DE MATURIDADE ORIENTADO POR EVIDÊNCIAS

Maturidade é tratada como capacidade demonstrável, não como quantidade de ferramentas. O Quadro 3 apresenta cinco níveis cumulativos. A progressão deve ser avaliada por serviços e riscos materiais; uma organização pode estar em níveis diferentes conforme o domínio analisado.

Quadro 3 – Níveis de maturidade e critérios verificáveis

Nível	Capacidade dominante	Evidência esperada	Critério de progressão
1 — Fragmentado	Inventário e responsabilidades incompletos	Registros dispersos e mudanças manuais	Identificar ativos críticos, responsáveis e fontes de verdade
2 — Declarado	Intenção versionada e revisão de mudanças	Repositórios, baselines e aprovações	Observar estado real e relacioná-lo à versão declarada

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.

Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

Nível	Capacidade dominante	Evidência esperada	Critério de progressão
3 — Rastreável	Proveniência entre revisão, build, artefato e implantação	Atestações, assinaturas, identidades e digests	Consultar a cadeia sob prazo e detectar lacunas
4 — Responsivo	Drift classificado e tratado por risco	Alertas priorizados, exceções com validade e reconciliações	Demonstrar tempos, cobertura, confiança e resultados
5 — Adaptativo	Controles aprendem com incidentes e métricas	Revisões de eficácia, testes de controle e mudanças transparentes	Reduzir reincidência sem aumentar atrito ou risco residual

Fonte: elaborado pelo autor.

A progressão não pressupõe automação integral. Ambientes simples, com responsabilidades claras e evidências confiáveis, podem ser mais governáveis do que stacks extensos e desconectados. Avaliações devem testar se a organização consegue reconstruir uma mudança, identificar exposição, restaurar estado e explicar decisões a uma parte independente.

No nível adaptativo, políticas são versionadas e testadas, exceções expiram, e mudanças nos controles passam por processo transparente. A plataforma oferece caminhos aprovados, mas permite contestação e evolução. Automação coleta, compara e executa respostas previsíveis; pessoas definem valores, aceitam riscos e revisam consequências.

8 DISCUSSÃO

8.1 Implicações teóricas e práticas

A principal implicação teórica é separar governabilidade de conformidade momentânea. Um snapshot conforme pode não explicar como o estado foi produzido; um desvio pode ser legítimo quando autorizado e temporário. O modelo desloca a unidade de análise do arquivo isolado para a cadeia entre decisão, artefato, atividade e observação. Isso aproxima governança, proveniência e operação sem afirmar que uma delas substitui as demais.

Na prática, a organização pode começar por perguntas verificáveis: qual artefato tem autoridade? Quem aprovou a versão em execução? A implantação deriva do build esperado? O estado observado está dentro das tolerâncias? A evidência pode ser recuperada no tempo necessário? Quando a resposta é desconhecida, a lacuna deve ser registrada como incerteza, e não preenchida por presunção.

O modelo também reduz o falso dilema entre velocidade e controle. Caminhos aprovados, políticas

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

testadas e evidências produzidas durante o fluxo evitam reconstrução manual posterior. Entretanto, controle mal desenhado aumenta carga cognitiva e incentiva rotas paralelas. A governança precisa ser consumível, observável e ajustável, além de tecnicamente rigorosa.

8.2 Limitações e agenda de pesquisa

O trabalho não valida o construto, não define pesos nem estabelece limiares de materialidade. As categorias podem apresentar sobreposição, e diferentes avaliadores podem discordar quanto a legitimidade, alcance e resposta. A literatura selecionada concentra-se em segurança, engenharia de software e normas internacionais; aspectos jurídicos, setoriais e culturais exigem estudos próprios.

A agenda de pesquisa inclui: operacionalizar dimensões sem ocultar seus componentes; medir concordância entre avaliadores; testar se completude de proveniência e tempo de evidência predizem qualidade de reconstrução; comparar detecção sintática e semântica; analisar custos e efeitos adversos da reconciliação automática; e realizar estudos multiorganizacionais que separem correlação de causalidade.

Também são necessários protocolos de refutação. O modelo perde utilidade se a classificação não melhorar priorização em relação a alertas técnicos simples, se a proveniência não reduzir tempo ou incerteza de investigação, se controles incorporados apenas deslocarem trabalho, ou se níveis de maturidade não puderem ser avaliados com concordância aceitável. Esses resultados devem ser tratados como evidência contra o modelo, não como falha de adoção.

9 CONCLUSÃO

Infraestruturas nativas de nuvem tornam insuficiente uma governança baseada apenas em avaliações periódicas. A divergência relevante não é qualquer diferença textual, mas o afastamento material entre intenção autorizada e realidade operacional, interpretado segundo risco, legitimidade, duração e responsabilidade. Governance drift oferece uma categoria analítica para examinar esse afastamento sem presumir que toda divergência constitua falha.

A síntese mostrou que intenção versionada, estado observado, proveniência consultável, comparação semântica e resposta proporcional formam capacidades complementares. Policy as code e pipelines DevSecOps ampliam prevenção, mas não eliminam riscos residuais nem substituem julgamento. A maturidade buscada é a capacidade de explicar, verificar e corrigir mudanças com evidências suficientes e incerteza explícita.

O modelo proposto permanece conceitual. Sua contribuição é fornecer fronteiras, etapas, classes e

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

critérios que podem ser contestados empiricamente. A próxima etapa científica é testar se essas capacidades melhoram reconstrução, priorização e recuperação em diferentes organizações, sem confundir maior volume de registros com melhor governança.

DECLARAÇÕES

Financiamento: não houve financiamento externo para este estudo.

Conflitos de interesse: o autor declara não possuir conflitos de interesse.

Disponibilidade de dados: o estudo não produziu conjunto de dados; as fontes analisadas estão identificadas nas referências.

Uso de inteligência artificial generativa: ferramenta de IA generativa da OpenAI foi utilizada como apoio editorial, linguístico e estrutural. O autor revisou o conteúdo, verificou as referências e assume responsabilidade integral pela versão submetida.

REFERÊNCIAS

BAETHGE, Christopher; GOLDBECK-WOOD, Sandra; MERTENS, Stephan. SANRA—a scale for the quality assessment of narrative review articles. Research Integrity and Peer Review, v. 4, art. 5, 2019. DOI: 10.1186/s41073-019-0064-8.

BEETZ, Florian; HARRER, Simon. GitOps: The Evolution of DevOps? IEEE Software, v. 39, n. 4, p. 70–75, 2022. DOI: 10.1109/MS.2021.3119106.

BURNS, Brendan; GRANT, Brian; OPPENHEIMER, David; BREWER, Eric; WILKES, John. Borg, Omega, and Kubernetes. Communications of the ACM, v. 59, n. 5, p. 50–57, 2016. DOI: 10.1145/2890784.

CNCF — CLOUD NATIVE COMPUTING FOUNDATION. CNCF Annual Survey 2023. San Francisco: CNCF, 2024. Disponível em: https://www.cncf.io/reports/cncf-annual-survey-2023/. Acesso em: 25 ago. 2026.

FORSGREN, Nicole; HUMBLE, Jez; KIM, Gene. Accelerate: The Science of Lean Software and DevOps. Portland: IT Revolution, 2018.

GOOGLE CLOUD; DORA. Accelerate State of DevOps 2024. Mountain View: Google Cloud, 2024. Disponível em: https://dora.dev/research/2024/dora-report/. Acesso em: 25 ago. 2026.

GUERRIERO, Michele; GARRIGA, Martin; TAMBURRI, Damian A.; PALOMBA, Fabio. Adoption, Support, and Challenges of Infrastructure-as-Code: Insights from Industry. In: IEEE INTERNATIONAL CONFERENCE ON SOFTWARE MAINTENANCE AND EVOLUTION, 2019. Proceedings [...]. IEEE, 2019. p. 580–589. DOI: 10.1109/ICSME.2019.00092.

HUMBLE, Jez; FARLEY, David. Continuous Delivery: Reliable Software Releases through Build,

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

Test, and Deployment Automation. Boston: Addison-Wesley, 2010.

ISO — INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO/IEC 27001:2022: Information security, cybersecurity and privacy protection — Information security management systems — Requirements. Geneva: ISO, 2022a.

ISO — INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO/IEC 27002:2022: Information security, cybersecurity and privacy protection — Information security controls. Geneva: ISO, 2022b.

NEWMAN, Zachary; MEYERS, John Speed; TORRES-ARIAS, Santiago. Sigstore: Software Signing for Everybody. In: ACM SIGSAC CONFERENCE ON COMPUTER AND COMMUNICATIONS SECURITY, 2022. Proceedings [...]. New York: ACM, 2022. p. 2353–2367. DOI: 10.1145/3548606.3560596.

NIST — NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. SP 800-137: Information Security Continuous Monitoring (ISCM) for Federal Information Systems and Organizations. Gaithersburg: NIST, 2011. DOI: 10.6028/NIST.SP.800-137.

NIST — NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. SP 800-128: Guide for Security-Focused Configuration Management of Information Systems. Atualização 1. Gaithersburg: NIST, 2019. DOI: 10.6028/NIST.SP.800-128.

NIST — NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. SP 800-53 Rev. 5: Security and Privacy Controls for Information Systems and Organizations. Gaithersburg: NIST, 2020. DOI: 10.6028/NIST.SP.800-53r5.

NIST — NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. SP 800-204C: Implementation of DevSecOps for a Microservices-based Application with Service Mesh. Gaithersburg: NIST, 2022a. DOI: 10.6028/NIST.SP.800-204C.

NIST — NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. SP 800-218: Secure Software Development Framework (SSDF) Version 1.1. Gaithersburg: NIST, 2022b. DOI: 10.6028/NIST.SP.800-218.

NIST — NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. SP 800-204D: Strategies for the Integration of Software Supply Chain Security in DevSecOps CI/CD Pipelines. Gaithersburg: NIST, 2024. DOI: 10.6028/NIST.SP.800-204D.

RAHMAN, Akond; MAHDAVI-HEZAVEH, Rezvan; WILLIAMS, Laurie. A systematic mapping study of infrastructure as code research. Information and Software Technology, v. 108, p. 65–77, 2019. DOI: 10.1016/j.infsof.2018.12.004.

SLSA COMMUNITY. Supply-chain Levels for Software Artifacts: Specification, version 1.2. 2025. Disponível em: https://slsa.dev/spec/v1.2/. Acesso em: 25 ago. 2026.

TORRES-ARIAS, Santiago; AFZALI, Hammad; KUPPUSAMY, Trishank Karthik; CURTMOLA, Reza; CAPPOS, Justin. in-toto: Providing farm-to-table guarantees for bits and bytes. In: USENIX SECURITY SYMPOSIUM, 28., 2019. Proceedings [...]. Berkeley: USENIX Association, 2019. p. 1393–1410.

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

RCMOS – Revista Científica Multidisciplinar O Saber.
ISSN: 2675-9128. São Paulo-SP.

Ano VII, v.2 2026 | submissão: 15/01/2026 | aceito: 20/01/2026 | publicação: 25/01/2026

W3C — WORLD WIDE WEB CONSORTIUM. PROV-O: The PROV Ontology. W3C Recommendation, 30 abr. 2013. Disponível em: https://www.w3.org/TR/prov-o/. Acesso em: 25 ago. 2026.

Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.