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

Ano VII, v.2 2026 | submissão: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

Centralização de regras de proteção de dados pessoais como estratégia de engenharia para a redução de defeitos de privacidade em sistemas corporativos

Centralization of personal data protection rules as an engineering strategy for reducing privacy defects in enterprise systems

Centralización de reglas de protección de datos personales como estrategia de ingeniería para la reducción de defectos de privacidad en sistemas corporativos

William Vitor Florencio de Abreu
Bacharel em Sistemas de Informação pela Universidade Metodista de São Paulo. Técnico em Tecnologia da Informação pela Microcamp.

RESUMO
O crescimento do volume de dados pessoais tratados por sistemas corporativos e a consolidação de marcos regulatórios como a Lei Geral de Proteção de Dados Pessoais e o Regulamento Geral sobre a Proteção de Dados europeu transformaram a privacidade em requisito estrutural da engenharia de software. Ao mesmo tempo, a adoção acelerada de assistentes de programação baseados em inteligência artificial ampliou a quantidade de código produzido e expôs a fragilidade de controles de privacidade que dependem da atenção individual de cada desenvolvedor. Este artigo analisa a centralização das regras de proteção de dados pessoais como estratégia arquitetural para reduzir defeitos de privacidade em sistemas corporativos. Trata-se de pesquisa qualitativa, exploratória, de natureza bibliográfica e documental, que articula legislação, normas técnicas, literatura de engenharia de privacidade e estudos empíricos recentes sobre segurança de código gerado por inteligência artificial. Os resultados indicam que a dispersão das regras de privacidade pelo código configura um problema de interesse transversal, associado a inconsistências, vazamentos por registros de log e dificuldade de demonstrar conformidade. Princípios clássicos de projeto seguro, como economia de mecanismo, mediação completa e padrões seguros por omissão, oferecem fundamento teórico para concentrar a definição das regras em um ponto único e auditável, preservando a aplicação distribuída. Conclui-se que a centralização favorece a consistência, a rastreabilidade e o uso mais seguro de ferramentas de inteligência artificial, desde que acompanhada de governança, testes automatizados e atenção aos riscos de ponto único de falha.
Palavras-chave: Proteção de dados pessoais. Privacidade desde a concepção. Arquitetura de software. LGPD. Inteligência artificial.

ABSTRACT
The growing volume of personal data processed by enterprise systems and the consolidation of regulatory frameworks such as the Brazilian General Data Protection Law and the European General Data Protection Regulation have turned privacy into a structural requirement of software engineering. At the same time, the rapid adoption of artificial intelligence coding assistants has increased the amount of code produced and exposed the weakness of privacy controls that depend on the individual attention of each developer. This article analyzes the centralization of personal data protection rules as an architectural strategy for reducing privacy defects in enterprise systems. It is a qualitative and exploratory study, based on bibliographic and documentary research, combining legislation, technical standards, privacy engineering literature and recent empirical studies on the security of AI-generated code. The findings indicate that scattering privacy rules across the codebase constitutes a crosscutting concern, associated with inconsistencies, leaks through log records and difficulty in demonstrating compliance. Classic secure design principles, such as economy of mechanism, complete mediation and fail-safe defaults, provide a theoretical basis for concentrating the definition of rules in a single auditable point while preserving distributed enforcement. The study concludes that centralization supports consistency, traceability and safer use of artificial intelligence tools, provided it is accompanied by governance, automated testing and attention to single point of failure risks.

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

Keywords: Personal data protection. Privacy by design. Software architecture. LGPD. Artificial intelligence.

1 INTRODUÇÃO

A digitalização de serviços públicos e privados elevou de forma expressiva o volume de dados pessoais armazenados, processados e compartilhados por sistemas corporativos. Cadastros de servidores, históricos acadêmicos, apólices de seguro e prontuários circulam por bancos de dados, interfaces de programação, filas de mensagens e ferramentas de observabilidade, e cada um desses pontos representa uma oportunidade de exposição indevida. A Lei Geral de Proteção de Dados Pessoais (BRASIL, 2018) e o Regulamento Geral sobre a Proteção de Dados da União Europeia (UNIÃO EUROPEIA, 2016) deslocaram essa preocupação do campo jurídico para o interior do processo de desenvolvimento, ao exigirem medidas técnicas aptas a proteger os dados desde a concepção do produto. O impacto econômico do descumprimento é concreto. Segundo o relatório Cost of a Data Breach, o custo médio de uma violação de dados no Brasil alcançou R$ 7,19 milhões em 2025, alta de 6,5% em relação aos R$ 6,75 milhões apurados no ano anterior (IBM, 2025).

A esse cenário soma-se a rápida incorporação de assistentes de programação baseados em inteligência artificial. A pesquisa anual do Stack Overflow registrou que 84% dos desenvolvedores utilizam ou pretendem utilizar essas ferramentas (STACK OVERFLOW, 2025), e um experimento controlado conduzido por Peng et al. (2023) observou que programadores apoiados por um assistente concluíram uma tarefa 55,8% mais rápido que o grupo de controle. A velocidade, contudo, não se traduz automaticamente em segurança. Pearce et al. (2022) verificaram que cerca de 40% dos programas gerados por um assistente em cenários de risco continham vulnerabilidades, e o relatório da Veracode (2025) identificou falhas de segurança em 45% das tarefas avaliadas em mais de cem modelos de linguagem. Quando as regras de proteção de dados pessoais estão espalhadas pelo código e dependem da lembrança de cada desenvolvedor, o aumento do volume de código produzido tende a multiplicar as oportunidades de erro.

Diante desse contexto, o presente artigo tem por objetivo analisar, à luz da legislação, de normas técnicas e da literatura científica, a centralização das regras de proteção de dados pessoais como estratégia arquitetural para a redução de defeitos de privacidade em sistemas corporativos. A questão que orienta o estudo é em que medida concentrar a definição dessas regras em um ponto único e auditável contribui para consistência, conformidade regulatória e uso mais seguro de ferramentas de inteligência artificial no desenvolvimento. O texto organiza-se em nove seções. Após esta introdução e a descrição da metodologia, examinam-se os fundamentos normativos da proteção de dados, a

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

distância entre os princípios de privacidade desde a concepção e a prática da engenharia, a dispersão das regras como problema arquitetural, os princípios de projeto que sustentam a centralização, os padrões de implementação e, por fim, a relação entre assistentes de inteligência artificial, defeitos de privacidade e custos de incidentes, seguida das conclusões.

2 METODOLOGIA

Este estudo adota abordagem qualitativa e caráter exploratório, adequados a temas em que o conhecimento acumulado ainda se encontra fragmentado entre áreas distintas e em que o objetivo é proporcionar maior familiaridade com o problema e construir hipóteses para investigações futuras (GIL, 2017). Quanto aos procedimentos técnicos, trata-se de pesquisa bibliográfica e documental. A dimensão bibliográfica reúne artigos científicos e livros de referência das áreas de engenharia de software, segurança da informação e engenharia de privacidade. A dimensão documental contempla textos legais, regulamentos da Autoridade Nacional de Proteção de Dados, normas técnicas internacionais, publicações do National Institute of Standards and Technology e relatórios setoriais com dados quantitativos sobre incidentes e sobre o uso de inteligência artificial no desenvolvimento de software.

A seleção das fontes seguiu procedimento inspirado nas diretrizes de revisão de literatura em engenharia de software propostas por Kitchenham e Charters (2007), sem a pretensão de constituir uma revisão sistemática completa. Privilegiaram-se trabalhos indexados em bases de reconhecida relevância para a computação, como IEEE Xplore, ACM Digital Library e SpringerLink, a partir de termos como privacy by design, privacy engineering, personally identifiable information, crosscutting concerns, secure design principles e AI code assistants security, em inglês e em português. A partir dos trabalhos considerados centrais, aplicou-se a técnica de bola de neve, que consiste em examinar as referências citadas por um estudo e os trabalhos que posteriormente o citaram, procedimento descrito por Wohlin (2014) como alternativa eficiente às buscas exclusivamente baseadas em palavras-chave.

Foram adotados como critérios de inclusão a pertinência direta ao tema, a publicação em periódicos, conferências ou instituições reconhecidas e a possibilidade de verificação pública do conteúdo. Obras clássicas, como os princípios de projeto seguro de Saltzer e Schroeder (1975) e o critério de decomposição modular de Parnas (1972), foram mantidas por constituírem fundamento teórico ainda vigente, independentemente da data de publicação. Relatórios produzidos por empresas privadas, como os da IBM e da Veracode, foram utilizados como indicadores de tendência e tratados com cautela, uma vez que refletem metodologias proprietárias, amostras específicas e potencial interesse

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

comercial de quem os publica.

A análise dos materiais ocorreu por meio de leitura analítica e síntese interpretativa, organizando os achados em eixos temáticos correspondentes às seções de desenvolvimento do artigo. Os dados numéricos apresentados foram extraídos diretamente das fontes citadas, sem extrapolações, e o gráfico incluído foi elaborado pelo autor a partir de valores publicados. Reconhece-se como limitação a ausência de coleta empírica própria, razão pela qual as conclusões têm caráter analítico e propositivo, devendo ser confirmadas por estudos de caso e experimentos controlados em ambientes organizacionais reais.

3 FUNDAMENTOS NORMATIVOS DA PROTEÇÃO DE DADOS PESSOAIS NO DESENVOLVIMENTO DE SOFTWARE

A Lei nº 13.709/2018 define dado pessoal como a informação relacionada a pessoa natural identificada ou identificável e estabelece uma categoria especial, a dos dados pessoais sensíveis, que abrange, entre outros, origem racial ou étnica, convicção religiosa, opinião política, filiação sindical, dados referentes à saúde ou à vida sexual e dados genéticos ou biométricos, quando vinculados a uma pessoa natural (BRASIL, 2018). Essa distinção tem consequência direta para a engenharia, porque implica que diferentes atributos de uma mesma entidade exigem graus distintos de proteção. Doneda (2019) analisa como a proteção de dados se consolidou como disciplina autônoma em relação ao direito à privacidade tradicional, deslocando o foco do segredo para o controle sobre o fluxo das informações. Essa trajetória culminou na Emenda Constitucional nº 115, que incluiu a proteção de dados pessoais entre os direitos e garantias fundamentais previstos no artigo 5º da Constituição Federal (BRASIL, 2022).

O artigo 6º da LGPD enumera os princípios que devem orientar qualquer tratamento, entre os quais finalidade, adequação, necessidade, transparência, segurança, prevenção e responsabilização e prestação de contas (BRASIL, 2018). Para o desenvolvedor de software, esses princípios não permanecem abstratos. O princípio da necessidade, que limita o tratamento ao mínimo indispensável para a finalidade pretendida, traduz-se em decisões sobre quais campos coletar, armazenar e exibir. O princípio da prevenção impõe a adoção de medidas que evitem danos antes que ocorram, e não apenas a reação a incidentes. Já o princípio da responsabilização exige que o agente de tratamento seja capaz de demonstrar a adoção de medidas eficazes, o que pressupõe que essas medidas estejam documentadas, rastreáveis e passíveis de verificação por terceiros.

A relação entre norma e arquitetura torna-se explícita nos artigos 46 e 49 da lei brasileira. O primeiro obriga os agentes de tratamento a adotar medidas de segurança, técnicas e administrativas, capazes

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

de proteger os dados pessoais de acessos não autorizados e de situações acidentais ou ilícitas, e seu parágrafo 2º determina que tais medidas sejam observadas desde a fase de concepção do produto ou do serviço até a sua execução. O artigo 49, por sua vez, exige que os sistemas utilizados no tratamento sejam estruturados de forma a atender aos requisitos de segurança e aos princípios gerais da lei (BRASIL, 2018). O Regulamento europeu segue linha equivalente ao prever, em seu artigo 25, a proteção de dados desde a concepção e por padrão e, no artigo 32, medidas como a pseudonimização e a cifragem (UNIÃO EUROPEIA, 2016).

A efetividade dessas exigências depende da capacidade de fiscalização e sanção. A LGPD prevê, no artigo 52, multa simples de até 2% do faturamento da pessoa jurídica de direito privado no Brasil, limitada a R$ 50 milhões por infração, além de sanções como publicização da infração e bloqueio dos dados (BRASIL, 2018). A Autoridade Nacional de Proteção de Dados regulamentou os critérios de cálculo dessas penalidades por meio da Resolução CD/ANPD nº 4, de 2023, e estabeleceu, na Resolução CD/ANPD nº 15, de 2024, o prazo de três dias úteis para a comunicação de incidentes de segurança que possam acarretar risco ou dano relevante aos titulares (AUTORIDADE NACIONAL DE PROTEÇÃO DE DADOS, 2023, 2024). No contexto europeu, o artigo 83 do Regulamento admite multas de até 20 milhões de euros ou 4% do faturamento anual mundial, prevalecendo o valor mais elevado (UNIÃO EUROPEIA, 2016).

Complementam esse arcabouço as normas técnicas que procuram traduzir princípios jurídicos em práticas organizacionais. A ISO/IEC 29100 estabelece um quadro de referência terminológico e principiológico para a privacidade em sistemas de tecnologia da informação (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2011), enquanto a ISO/IEC 27701 estende os sistemas de gestão de segurança da informação para abranger a gestão de informações de privacidade (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2019). O guia NIST SP 800-122 propõe classificar as informações pessoalmente identificáveis segundo níveis de impacto de confidencialidade baixo, moderado e alto, recomendando a minimização do uso e a desidentificação sempre que possível (MCCALLISTER; GRANCE; SCARFONE, 2010). Bioni (2019) observa que o consentimento do titular, isoladamente, é insuficiente para assegurar proteção efetiva, o que reforça a importância de salvaguardas estruturais. Nenhum desses instrumentos prescreve uma arquitetura específica, mas todos convergem para a exigência de controles consistentes, documentados e demonstráveis.

4 PRIVACIDADE DESDE A CONCEPÇÃO E A DISTÂNCIA ENTRE PRINCÍPIO 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.

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

Ano VII, v.2 2026 | submissão: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

PRÁTICA

A expressão privacidade desde a concepção foi sistematizada por Ann Cavoukian, então comissária de informação e privacidade da província de Ontário, no Canadá, em um conjunto de sete princípios fundamentais. São eles a postura proativa e não reativa, a privacidade como configuração padrão, a privacidade incorporada ao projeto, a funcionalidade total em lógica de soma positiva, a segurança de ponta a ponta durante todo o ciclo de vida dos dados, a visibilidade e transparência e o respeito à privacidade do usuário (CAVOUKIAN, 2009). A influência desse trabalho é visível na redação do artigo 25 do Regulamento europeu e no parágrafo 2º do artigo 46 da LGPD, que incorporaram a ideia de que a proteção deve nascer com o sistema, e não ser acrescentada posteriormente como camada externa de conformidade.

Apesar de sua ampla aceitação, os princípios de Cavoukian receberam críticas quanto à dificuldade de operacionalização. Gürses, Troncoso e Diaz (2011) argumentam que a privacidade desde a concepção permanece vaga enquanto não for acompanhada de métodos de engenharia capazes de orientar decisões técnicas concretas, como a escolha entre arquiteturas que concentram ou que minimizam os dados coletados. Spiekermann e Cranor (2009) já haviam proposto distinguir duas abordagens complementares. A privacidade por política depende de avisos, termos e escolhas do usuário, enquanto a privacidade por arquitetura reduz a necessidade de confiança ao limitar, por construção, a identificabilidade e a exposição dos dados. Para os autores, a segunda abordagem oferece garantias mais robustas, justamente por não depender do comportamento correto de cada participante do processo.

Na tentativa de aproximar princípios e prática, Hoepman (2014) propôs oito estratégias de projeto de privacidade, divididas entre orientadas a dados e orientadas a processos. As estratégias minimizar, ocultar, separar e agregar tratam da forma como os dados são coletados e armazenados, ao passo que informar, controlar, impor e demonstrar dizem respeito à governança do tratamento e à prestação de contas. A contribuição do autor reside em oferecer uma camada intermediária entre a norma jurídica e o padrão de projeto de software, permitindo que equipes de engenharia associem requisitos legais a decisões arquiteturais verificáveis. As estratégias impor e demonstrar são particularmente relevantes para este estudo, pois pressupõem mecanismos capazes de aplicar políticas de modo uniforme e de comprovar essa aplicação perante auditorias e autoridades.

A literatura empírica revela, contudo, que a distância entre princípio e prática persiste no cotidiano das equipes. Hadar et al. (2018), em estudo qualitativo com desenvolvedores de software, constataram que esses profissionais tendem a interpretar a privacidade sob a lente da segurança da informação, reduzindo-a, com frequência, a uma questão de proteção técnica dos dados, e que o clima

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

organizacional exerce influência significativa sobre a atenção dedicada ao tema. Senarath e Arachchilage (2018), ao solicitarem a desenvolvedores que incorporassem requisitos de privacidade a um projeto, observaram, entre os 36 participantes, dificuldades em relacionar esses requisitos a decisões de implementação e insegurança quanto à correção das soluções adotadas. Apenas 11 deles incorporaram a pseudonimização a seus projetos, embora a maioria declarasse aplicar a minimização de dados. Os dois estudos sugerem que a proteção efetiva não pode depender exclusivamente do conhecimento individual de quem escreve o código.

Desses achados decorre uma implicação relevante para o desenho de sistemas. Se a aplicação correta das regras de privacidade exige que cada desenvolvedor, em cada nova funcionalidade, identifique quais dados são sensíveis, recorde as regras aplicáveis e as implemente sem erro, a probabilidade de falha tende a crescer com o número de pontos de decisão e ao número de pessoas envolvidas. Em equipes grandes, com alta rotatividade ou com participação de fornecedores externos, essa dependência torna-se ainda mais frágil. A resposta coerente com a proposta de Spiekermann e Cranor (2009) consiste em transferir parte da responsabilidade do indivíduo para a estrutura, fazendo com que o comportamento protetor seja propriedade do sistema e não resultado de escolhas repetidas, tema desenvolvido nas seções seguintes.

5 A DISPERSÃO DAS REGRAS DE PRIVACIDADE COMO PROBLEMA ARQUITETURAL

A engenharia de software reconhece desde a década de 1970 a importância da separação de interesses, expressão associada a Dijkstra (1982) para designar a prática de tratar isoladamente cada aspecto de um problema complexo. Alguns interesses, no entanto, resistem à decomposição tradicional em módulos, porque afetam simultaneamente muitas partes do sistema. Kiczales et al. (1997) denominaram esses casos de interesses transversais, cuja implementação tende a ficar espalhada e entrelaçada com o restante do código, e propuseram a programação orientada a aspectos para tratá-los de forma modular. A proteção de dados pessoais apresenta exatamente essa natureza. Ela incide sobre a persistência, sobre as interfaces de programação, sobre a geração de relatórios, sobre as exportações de arquivos, sobre as telas exibidas aos usuários e sobre os próprios registros de log.

Quando um interesse transversal não recebe tratamento arquitetural específico, ele tende a se manifestar como lógica duplicada e inconsistente. Em sistemas corporativos, essa situação pode traduzir-se em funções de mascaramento de documentos reimplementadas em módulos distintos, em verificações de permissão escritas de forma ad hoc em cada controlador e em listas de campos sensíveis mantidas manualmente em diferentes partes do código. Fowler (2018) descreve um sintoma

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

denominado shotgun surgery, em tradução livre cirurgia com espingarda, que ocorre quando uma única alteração conceitual exige modificações pequenas em muitos lugares diferentes. No caso da privacidade, uma mudança regulatória ou uma nova orientação da autoridade nacional pode demandar ajustes em dezenas de pontos, e basta que um deles seja esquecido para que o sistema passe a expor dados de forma indevida.

Os registros de log merecem atenção particular, porque podem se converter em canal de vazamento pouco vigiado. Mensagens de depuração podem serializar objetos inteiros, incluindo nomes, documentos e endereços, que depois são eventualmente replicados para plataformas de monitoramento, retidos por longos períodos e acessados por profissionais sem necessidade de conhecer aqueles dados. A lista OWASP Top 10 de 2021 inclui as falhas de registro e monitoramento de segurança entre os riscos mais críticos de aplicações web e mantém as falhas criptográficas, antes denominadas exposição de dados sensíveis, em posição de destaque (OWASP FOUNDATION, 2021). O relatório da Veracode (2025) acrescenta um dado preocupante ao indicar que os modelos de linguagem avaliados falharam em proteger o código contra injeção em logs em 88% dos casos testados.

A dispersão também afeta o momento em que os defeitos são descobertos. Boehm e Basili (2001) registraram que encontrar e corrigir um problema de software após a entrega é, com frequência, cem vezes mais caro do que fazê-lo nas fases de requisitos e projeto. Em matéria de privacidade, o custo de um defeito tardio não se limita ao retrabalho técnico. Uma exposição detectada em produção pode configurar incidente de segurança, com obrigação de comunicação à autoridade e aos titulares nos termos da regulamentação vigente, além de danos reputacionais. O relatório da IBM (2025) indica que o ciclo de vida médio de uma violação, somando identificação e contenção, foi de 241 dias em escala global, intervalo durante o qual dados expostos por uma falha silenciosa podem continuar circulando sem que a organização perceba.

Há, por fim, um efeito direto sobre a prestação de contas exigida pelo artigo 6º da LGPD. Demonstrar que um sistema protege adequadamente os dados pessoais pressupõe saber onde esses dados são tratados e quais regras se aplicam a cada um deles. Quando as regras estão espalhadas, esse inventário depende de inspeção manual do código, sujeita a omissões e rapidamente desatualizada. Parnas (1972) propôs que módulos fossem definidos de modo a ocultar decisões de projeto com maior probabilidade de mudança, de forma que alterações futuras afetassem o menor número possível de componentes. As regras de privacidade, sujeitas a revisões legais, regulatórias e contratuais, enquadram-se nesse critério e constituem candidatas naturais ao encapsulamento em um ponto único do sistema.

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

6 PRINCÍPIOS DE PROJETO QUE FUNDAMENTAM A CENTRALIZAÇÃO

Saltzer e Schroeder (1975), em trabalho que permanece referência obrigatória na área de segurança computacional, formularam oito princípios de projeto para mecanismos de proteção da informação. São eles a economia de mecanismo, os padrões seguros por omissão, a mediação completa, o projeto aberto, a separação de privilégios, o menor privilégio, o mecanismo menos comum e a aceitabilidade psicológica. O primeiro princípio sustenta que o mecanismo de proteção deve ser tão simples e pequeno quanto possível, porque apenas mecanismos compactos podem ser inspecionados linha a linha com razoável confiança. Uma regra de privacidade implementada uma única vez, em componente dedicado, atende a esse princípio de modo mais aderente do que dezenas de versões ligeiramente diferentes da mesma regra distribuídas pela base de código.

O princípio da mediação completa determina que todo acesso a um objeto protegido seja verificado quanto à autorização, sem exceções e sem confiança em verificações anteriores (SALTZER; SCHROEDER, 1975). Em uma arquitetura dispersa, a garantia de mediação completa depende de que nenhum desenvolvedor esqueça de chamar a verificação adequada, condição difícil de assegurar em sistemas extensos. Já o princípio dos padrões seguros por omissão estabelece que as decisões de acesso devem se basear em permissão explícita, de modo que a ausência de regra resulte em negação. Aplicado à privacidade, esse princípio sugere que um novo campo ainda não classificado seja tratado como sensível até que alguém declare o contrário, lógica alinhada à privacidade como configuração padrão defendida por Cavoukian (2009) e ao artigo 25 do Regulamento europeu.

O princípio do menor privilégio encontra expressão prática nos modelos de controle de acesso. O controle baseado em papéis, formalizado por Ferraiolo e Kuhn (1992), associa permissões a funções organizacionais e simplifica a administração em ambientes com muitos usuários. Para dados pessoais, porém, a decisão de exibir, mascarar ou ocultar uma informação frequentemente depende de atributos adicionais, como a classificação do dado, a finalidade do acesso, o vínculo entre o solicitante e o titular e o canal utilizado. O controle de acesso baseado em atributos, descrito no guia NIST SP 800-162, contempla esse tipo de decisão e propõe uma arquitetura em que pontos de aplicação de política consultam um ponto de decisão de política, separando a lógica de autorização da lógica de negócio (HU et al., 2014).

As estratégias de Hoepman (2014) encontram na centralização um meio particularmente adequado de operacionalização. A estratégia ocultar se concretiza em serviços únicos de mascaramento e pseudonimização. A estratégia separar orienta a segregação de dados sensíveis em repositórios ou esquemas próprios. A estratégia impor exige que a política seja aplicada de forma uniforme, o que se torna verificável quando existe um único ponto de definição. A estratégia demonstrar, por sua vez,

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

beneficia-se da possibilidade de gerar evidências automáticas sobre quais regras estão vigentes, quais campos estão classificados e quais acessos foram concedidos. Em todos esses casos, a concentração das regras transforma obrigações difusas em propriedades inspecionáveis do sistema, aproximando a engenharia da exigência de responsabilização prevista na legislação.

A centralização, contudo, não está isenta de riscos, e a própria obra de Saltzer e Schroeder (1975) oferece o contraponto ao enunciar o princípio do mecanismo menos comum, segundo o qual mecanismos compartilhados por vários usuários devem ser minimizados, uma vez que cada mecanismo comum representa um caminho potencial de fluxo indevido de informações e um ponto do qual todos passam a depender. Um componente central de privacidade mal projetado pode tornar-se ponto único de falha, gargalo de desempenho ou alvo preferencial de ataques. Bass, Clements e Kazman (2012) lembram que decisões arquiteturais envolvem sempre compromissos entre atributos de qualidade, como desempenho, disponibilidade, modificabilidade e segurança. A solução proposta pela literatura de controle de acesso, e adotada neste estudo, consiste em centralizar a definição das regras, preservando a aplicação distribuída por meio de bibliotecas, interceptadores e pontos de aplicação próximos a cada serviço, o que reduz o acoplamento operacional sem sacrificar a consistência.

7 PADRÕES DE IMPLEMENTAÇÃO DA CENTRALIZAÇÃO DE REGRAS DE DADOS PESSOAIS

O primeiro passo para centralizar regras de privacidade consiste em construir um catálogo de classificação dos dados, no qual cada atributo do modelo de domínio recebe metadados que indicam sua categoria. Identificadores diretos, como números de documentos, diferem de quase identificadores, como data de nascimento, gênero e código postal, cuja combinação pode permitir a reidentificação de pessoas. Sweeney (2002) demonstrou esse risco ao propor o modelo de k-anonimato, concebido justamente porque bases supostamente anônimas podiam ser reidentificadas por meio do cruzamento de atributos aparentemente inofensivos. O catálogo pode ainda incorporar os níveis de impacto sugeridos pelo NIST SP 800-122 (MCCALLISTER; GRANCE; SCARFONE, 2010), servindo como fonte única a partir da qual as demais camadas derivam seu comportamento.

Uma forma de tornar a classificação parte da linguagem do sistema é representá-la no próprio sistema de tipos. Evans (2003), ao descrever o projeto orientado a domínio, propõe os objetos de valor como representações de conceitos do negócio que carregam regras próprias de validação e comportamento. Aplicada à privacidade, essa ideia permite que um número de documento deixe de ser uma cadeia de caracteres genérica e passe a ser um tipo que sabe se mascarar, se serializar de forma segura e se

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

recusar a aparecer integralmente em contextos não autorizados. Sobre essa base, uma camada de política declarativa define, em um único repositório versionado, quais perfis podem visualizar cada categoria de dado e para quais finalidades, enquanto os pontos de aplicação consultam essa política em vez de reimplementá-la.

A camada central deve oferecer também serviços padronizados de transformação. O mascaramento parcial atende a cenários de exibição em que o usuário precisa reconhecer o dado sem visualizá-lo por completo. A pseudonimização, definida no artigo 4º do Regulamento europeu e no parágrafo 4º do artigo 13 da LGPD como o tratamento por meio do qual um dado perde a possibilidade de associação, direta ou indireta, a um indivíduo, senão pelo uso de informação adicional mantida separadamente, é útil em análises internas e ambientes de teste (UNIÃO EUROPEIA, 2016; BRASIL, 2018). Para estatísticas agregadas, técnicas como a privacidade diferencial, formalizada por Dwork (2006), oferecem garantias matemáticas contra a inferência de informações individuais. Concentrar essas técnicas em um serviço comum reduz o risco de soluções improvisadas e facilita a revisão especializada de sua correção.

Os pontos de saída do sistema exigem tratamento automatizado, já que é por eles que os dados deixam o perímetro controlado da aplicação. Interceptadores de serialização podem aplicar as regras do catálogo a toda resposta de interface de programação e a todo registro de log, de modo que um campo classificado como sensível seja mascarado por padrão, mesmo que o desenvolvedor não tenha tomado nenhuma providência específica. Esse mecanismo reproduz, em escala de aplicação, a lógica dos interesses transversais descrita por Kiczales et al. (1997). Em direção semelhante, Enck et al. (2010) demonstraram com o sistema TaintDroid a viabilidade de rastrear a propagação de dados sensíveis ao longo da execução de aplicativos, ideia que pode inspirar a marcação de valores derivados de dados pessoais para que a proteção os acompanhe durante todo o fluxo.

A centralização só cumpre sua finalidade se for acompanhada de mecanismos de verificação contínua. Testes automatizados podem afirmar, para cada categoria do catálogo, que os dados correspondentes não aparecem em logs, respostas ou exportações sem a transformação devida. Regras de análise estática podem sinalizar o acesso direto a atributos sensíveis fora da camada autorizada. McGraw (2006) defende que a segurança seja incorporada ao longo de todo o ciclo de desenvolvimento por meio de pontos de controle como revisão de código, análise de risco arquitetural e testes baseados em risco, perspectiva que se aplica integralmente à privacidade. Indicadores como a quantidade de defeitos relacionados a dados pessoais e o tempo necessário para proteger um novo campo permitem acompanhar a evolução do sistema e produzir evidências úteis para a prestação de contas.

8 ASSISTENTES DE INTELIGÊNCIA ARTIFICIAL, DEFEITOS DE PRIVACIDADE 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.

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

Ano VII, v.2 2026 | submissão: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

ECONOMIA DOS INCIDENTES

A adoção de ferramentas de inteligência artificial no desenvolvimento de software deixou de ser experimental. Segundo a pesquisa do Stack Overflow (2025), 84% dos respondentes utilizam ou planejam utilizar essas ferramentas, contra 76% no ano anterior, e cerca de metade dos desenvolvedores profissionais recorre a elas diariamente. A mesma pesquisa revela, no entanto, um paradoxo. A proporção de desenvolvedores que desconfiam da precisão das respostas geradas, de 46%, supera a dos que confiam nelas, de 33%. O ganho de produtividade, por sua vez, encontra respaldo experimental. No estudo de Peng et al. (2023), os participantes com acesso a um assistente de programação concluíram a implementação de um servidor HTTP em tempo 55,8% menor que o grupo de controle, resultado que ajuda a explicar a velocidade com que essas ferramentas vêm sendo incorporadas às rotinas corporativas.

Os estudos sobre a segurança do código gerado apontam limitações relevantes. Pearce et al. (2022) avaliaram 89 cenários associados a fraquezas conhecidas de software e verificaram que aproximadamente 40% dos 1.689 programas produzidos por um assistente continham vulnerabilidades. Perry et al. (2023), em estudo com usuários, observaram que participantes com acesso a um assistente escreveram código significativamente menos seguro e, ao mesmo tempo, mostraram-se mais propensos a acreditar que haviam produzido código seguro. O relatório da Veracode (2025), que testou mais de cem modelos de linguagem, encontrou vulnerabilidades em 45% das tarefas avaliadas e taxas de falha de 86% na proteção contra cross-site scripting e de 88% contra injeção em logs, categorias diretamente associadas à exposição de dados pessoais.

Esses resultados ajudam a compreender por que a centralização das regras de privacidade ganha relevância em equipes que utilizam inteligência artificial. Os modelos de linguagem geram código plausível no contexto imediato, mas, salvo quando explicitamente informados, desconhecem a classificação dos dados de cada organização, as finalidades autorizadas e as orientações regulatórias específicas. Quando a proteção depende de que o código gerado invoque corretamente funções de mascaramento espalhadas pelo sistema, cada sugestão aceita representa uma nova oportunidade de omissão. Quando, ao contrário, a proteção é propriedade da plataforma, expressa em tipos, interceptadores e políticas centrais, o código gerado tende a herdar o comportamento seguro por construção. A centralização estreita o espaço de erro possível e permite que a revisão humana se concentre nas alterações de política, onde o julgamento especializado é indispensável.

A dimensão econômica reforça esse argumento, ainda que exija interpretação cuidadosa. Os dados brasileiros do relatório da IBM (2025) indicam que organizações com uso extensivo de inteligência artificial e automação em suas operações de segurança registraram custo médio de R$ 6,48 milhões

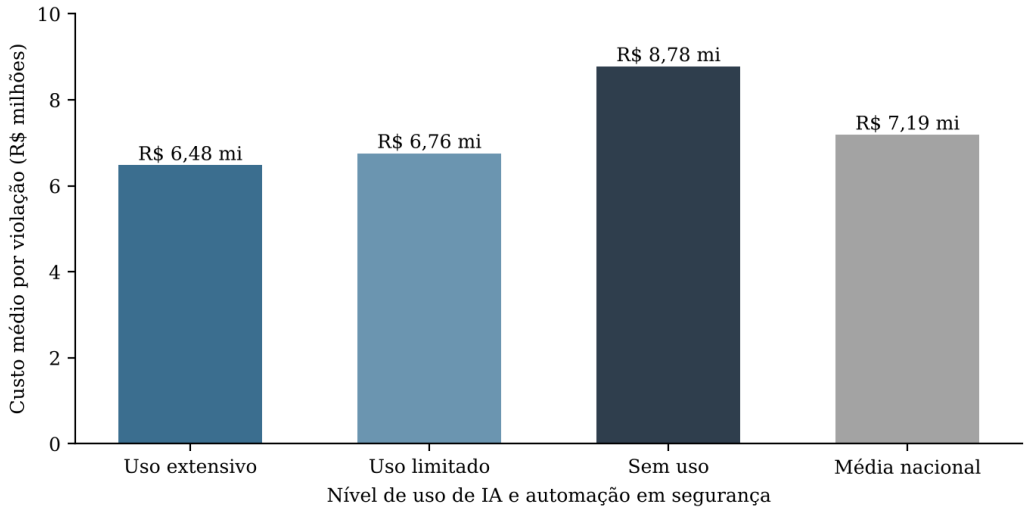
Este é um artigo publicado em acesso aberto (Open Access) sob a licença CreativeCommons Attribution, que permite uso, distribuição e reprodução em qualquer meio, sem restrições desde que o trabalho original seja corretamente citado.

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

Ano VII, v.2 2026 | submissão: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

por violação, inferior aos R$ 6,76 milhões observados entre organizações com uso limitado e aos R$ 8,78 milhões das que não utilizavam esses recursos, conforme sintetiza a Figura 1. O mesmo relatório estimou que o uso não autorizado de ferramentas de inteligência artificial, fenômeno denominado shadow AI, acrescentou em média R$ 591,4 mil ao custo dos incidentes no país. Tais números descrevem associações observadas em uma amostra específica e não estabelecem relação causal, mas são compatíveis com a hipótese de que a forma como a tecnologia é governada influencia o resultado econômico, uma vez que o uso não supervisionado aparece associado a custos adicionais.

Figura 1 – Custo médio de uma violação de dados no Brasil segundo o nível de uso de inteligência artificial e automação em segurança (2025)



Fonte: elaborado pelo autor com base em IBM (2025).

O enfrentamento desse cenário exige governança além da arquitetura. O NIST Privacy Framework organiza as atividades de gestão de riscos de privacidade nas funções identificar, governar, controlar, comunicar e proteger, oferecendo uma estrutura útil para integrar o uso de inteligência artificial às políticas organizacionais (NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY, 2020). Diretrizes internas podem definir quais ferramentas estão aprovadas, proibir o envio de dados reais de titulares em instruções aos modelos, exigir bases sintéticas em ambientes de teste e estabelecer revisão humana obrigatória para alterações nas políticas centrais. Cabe reconhecer que grande parte das evidências disponíveis provém de experimentos controlados e de relatórios de empresas do setor, o que torna necessária a realização de estudos empíricos em ambientes corporativos para medir com rigor o efeito da centralização sobre a incidência de defeitos.

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

9 CONCLUSÃO

Este artigo analisou a centralização das regras de proteção de dados pessoais como estratégia arquitetural para reduzir defeitos de privacidade em sistemas corporativos. A investigação partiu da constatação de que a legislação brasileira e a europeia exigem medidas técnicas desde a concepção dos sistemas, mas não indicam como estruturá-las, e de que a adoção acelerada de assistentes de inteligência artificial ampliou o volume de código produzido sem elevar, na mesma proporção, sua segurança. O problema, portanto, situa-se na interseção entre direito, arquitetura de software e novas práticas de produção de código.

A análise dos fundamentos normativos mostrou que a LGPD, o Regulamento europeu e as normas técnicas convergem na exigência de controles consistentes, documentados e demonstráveis. Os princípios da necessidade, da prevenção e da responsabilização, somados aos artigos 46 e 49 da lei brasileira, transformam a privacidade em requisito estrutural, cuja comprovação depende de evidências técnicas e não apenas de declarações formais de conformidade. O regime sancionatório regulamentado pela autoridade nacional reforça a urgência de tratar o tema como prioridade de engenharia.

A revisão da literatura sobre privacidade desde a concepção evidenciou a distância persistente entre princípios amplamente aceitos e a prática das equipes de desenvolvimento. Estudos empíricos indicam que desenvolvedores tendem a reduzir a privacidade a questões de segurança e encontram dificuldade em traduzir requisitos legais em decisões de implementação, o que torna arriscado depender exclusivamente do conhecimento individual para assegurar a proteção dos titulares. A abordagem de privacidade por arquitetura surge, nesse quadro, como resposta mais robusta do que a privacidade baseada apenas em políticas e orientações.

Verificou-se também que a proteção de dados pessoais constitui um interesse transversal, que incide sobre persistência, interfaces, relatórios, exportações e registros de log. Quando esse interesse não recebe tratamento arquitetural próprio, manifesta-se como lógica duplicada e inconsistente, dificulta a adaptação a mudanças regulatórias, favorece vazamentos silenciosos e compromete a capacidade da organização de demonstrar conformidade perante auditorias e autoridades. O custo crescente de defeitos descobertos tardiamente agrava esse quadro, sobretudo quando envolvem dados de titulares e configuram incidentes sujeitos a comunicação obrigatória.

Os princípios de projeto seguro formulados por Saltzer e Schroeder, especialmente a economia de mecanismo, a mediação completa e os padrões seguros por omissão, oferecem fundamento teórico sólido para a centralização. Combinados às estratégias de Hoepman e aos modelos de controle de acesso baseados em atributos, esses princípios permitem converter obrigações difusas em

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

propriedades verificáveis do sistema, alinhando a engenharia à exigência legal de responsabilização. A concentração das regras favorece ainda a adaptação a mudanças normativas, pois uma alteração passa a exigir intervenção em um único ponto.

No plano da implementação, o estudo identificou um conjunto coerente de padrões que inclui catálogo de classificação de dados, representação da sensibilidade no sistema de tipos, camada de política declarativa e versionada, serviços comuns de mascaramento e pseudonimização, interceptadores de serialização e registro, além de testes automatizados e análise estática. A combinação desses elementos torna o comportamento protetor o padrão do sistema, reduzindo a dependência de decisões individuais e produzindo evidências contínuas que sustentam a prestação de contas exigida pela legislação.

Em relação à inteligência artificial, as evidências disponíveis indicam que o código gerado por modelos de linguagem apresenta taxas relevantes de vulnerabilidade, inclusive em categorias associadas à exposição de dados pessoais. A centralização mostra-se especialmente oportuna nesse contexto, porque tende a fazer com que o código produzido com auxílio dessas ferramentas herde a proteção por construção, desde que acompanhada de governança clara sobre o uso de dados reais e de revisão humana das políticas. Esse quadro indica que a forma de uso dessas ferramentas é tão relevante quanto a própria tecnologia empregada.

Reconhece-se que a centralização envolve compromissos, como o risco de ponto único de falha e possíveis impactos sobre o desempenho, mitigáveis pela separação entre definição central e aplicação distribuída das regras. Por se tratar de estudo bibliográfico e documental, recomenda-se que pesquisas futuras realizem estudos de caso e experimentos em ambientes corporativos, medindo a incidência de defeitos de privacidade antes e depois da adoção da abordagem e avaliando seus efeitos em equipes que utilizam assistentes de inteligência artificial, de modo a consolidar evidências empíricas sobre os benefícios aqui discutidos.

REFERÊNCIAS

AUTORIDADE NACIONAL DE PROTEÇÃO DE DADOS (Brasil). **Resolução CD/ANPD nº 4, de 24 de fevereiro de 2023**. Aprova o Regulamento de Dosimetria e Aplicação de Sanções Administrativas. Brasília, DF: ANPD, 2023.

AUTORIDADE NACIONAL DE PROTEÇÃO DE DADOS (Brasil). **Resolução CD/ANPD nº 15, de 24 de abril de 2024**. Aprova o Regulamento de Comunicação de Incidente de Segurança. Brasília, DF: ANPD, 2024.

BASS, Len; CLEMENTS, Paul; KAZMAN, Rick. **Software architecture in practice**. 3. ed. Upper Saddle River: Addison-Wesley, 2012.

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

BIONI, Bruno Ricardo. **Proteção de dados pessoais**: a função e os limites do consentimento. Rio de Janeiro: Forense, 2019.

BOEHM, Barry; BASILI, Victor R. Software defect reduction top 10 list. **Computer**, Los Alamitos, v. 34, n. 1, p. 135-137, jan. 2001.

BRASIL. Emenda Constitucional nº 115, de 10 de fevereiro de 2022. Altera a Constituição Federal para incluir a proteção de dados pessoais entre os direitos e garantias fundamentais e para fixar a competência privativa da União para legislar sobre proteção e tratamento de dados pessoais. **Diário Oficial da União**, Brasília, DF, 11 fev. 2022.

BRASIL. Lei nº 13.709, de 14 de agosto de 2018. Lei Geral de Proteção de Dados Pessoais (LGPD). **Diário Oficial da União**, Brasília, DF, 15 ago. 2018.

CAVOUKIAN, Ann. **Privacy by design**: the 7 foundational principles. Toronto: Information and Privacy Commissioner of Ontario, 2009.

DIJKSTRA, Edsger W. On the role of scientific thought. In: DIJKSTRA, Edsger W. **Selected writings on computing**: a personal perspective. New York: Springer, 1982. p. 60-66.

DONEDA, Danilo. **Da privacidade à proteção de dados pessoais**: elementos da formação da Lei Geral de Proteção de Dados. 2. ed. São Paulo: Thomson Reuters Brasil, 2019.

DWORK, Cynthia. Differential privacy. In: INTERNATIONAL COLLOQUIUM ON AUTOMATA, LANGUAGES AND PROGRAMMING, 33., 2006, Venice. **Proceedings** [...]. Berlin: Springer, 2006. p. 1-12. (Lecture Notes in Computer Science, v. 4052).

ENCK, William et al. TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones. In: USENIX SYMPOSIUM ON OPERATING SYSTEMS DESIGN AND IMPLEMENTATION, 9., 2010, Vancouver. **Proceedings** [...]. Berkeley: USENIX Association, 2010. p. 393-407.

EVANS, Eric. **Domain-driven design**: tackling complexity in the heart of software. Boston: Addison-Wesley, 2003.

FERRAIOLO, David F.; KUHN, D. Richard. Role-based access controls. In: NATIONAL COMPUTER SECURITY CONFERENCE, 15., 1992, Baltimore. **Proceedings** [...]. Baltimore: NIST; NCSC, 1992. p. 554-563.

FOWLER, Martin. **Refactoring**: improving the design of existing code. 2. ed. Boston: Addison-Wesley, 2018.

GIL, Antonio Carlos. **Como elaborar projetos de pesquisa**. 6. ed. São Paulo: Atlas, 2017.

GÜRSES, Seda; TRONCOSO, Carmela; DIAZ, Claudia. Engineering privacy by design. In: COMPUTERS, PRIVACY & DATA PROTECTION CONFERENCE, 4., 2011, Bruxelas. **Anais** [...]. Bruxelas: CPDP, 2011.

HADAR, Irit et al. Privacy by designers: software developers' privacy mindset. **Empirical Software Engineering**, New York, v. 23, n. 1, p. 259-289, 2018.

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

HOEPMAN, Jaap-Henk. Privacy design strategies. In: IFIP INTERNATIONAL INFORMATION SECURITY CONFERENCE, 29., 2014, Marrakech. **ICT systems security and privacy protection**. Berlin: Springer, 2014. p. 446-459. (IFIP Advances in Information and Communication Technology, v. 428).

HU, Vincent C. et al. **Guide to attribute based access control (ABAC) definition and considerations**. Gaithersburg: National Institute of Standards and Technology, 2014. (NIST Special Publication 800-162).

IBM. **Cost of a data breach report 2025**. Armonk: IBM Corporation, 2025.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. **ISO/IEC 29100**: information technology, security techniques, privacy framework. Geneva: ISO, 2011.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. **ISO/IEC 27701**: security techniques, extension to ISO/IEC 27001 and ISO/IEC 27002 for privacy information management, requirements and guidelines. Geneva: ISO, 2019.

KICZALES, Gregor et al. Aspect-oriented programming. In: EUROPEAN CONFERENCE ON OBJECT-ORIENTED PROGRAMMING, 11., 1997, Jyväskylä. **Proceedings** [...]. Berlin: Springer, 1997. p. 220-242. (Lecture Notes in Computer Science, v. 1241).

KITCHENHAM, Barbara; CHARTERS, Stuart. **Guidelines for performing systematic literature reviews in software engineering**. Keele: Keele University; Durham: Durham University, 2007. (EBSE Technical Report, EBSE-2007-01).

MCCALLISTER, Erika; GRANCE, Tim; SCARFONE, Karen. **Guide to protecting the confidentiality of personally identifiable information (PII)**. Gaithersburg: National Institute of Standards and Technology, 2010. (NIST Special Publication 800-122).

MCGRAW, Gary. **Software security**: building security in. Upper Saddle River: Addison-Wesley, 2006.

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. **NIST privacy framework**: a tool for improving privacy through enterprise risk management, version 1.0. Gaithersburg: NIST, 2020.

OWASP FOUNDATION. **OWASP Top 10**: 2021. [S. l.]: OWASP Foundation, 2021. Disponível em: https://owasp.org/Top10/.

PARNAS, David L. On the criteria to be used in decomposing systems into modules. **Communications of the ACM**, New York, v. 15, n. 12, p. 1053-1058, dez. 1972.

PEARCE, Hammond et al. Asleep at the keyboard? Assessing the security of GitHub Copilot's code contributions. In: IEEE SYMPOSIUM ON SECURITY AND PRIVACY, 43., 2022, San Francisco. **Proceedings** [...]. Los Alamitos: IEEE, 2022. p. 754-768.

PENG, Sida et al. **The impact of AI on developer productivity**: evidence from GitHub Copilot. arXiv, 2023. Disponível em: https://arxiv.org/abs/2302.06590.

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: 02/05/2026 | aceito: 04/06/2026 | publicação: 22/07/2026

PERRY, Neil et al. Do users write more insecure code with AI assistants? In: ACM SIGSAC CONFERENCE ON COMPUTER AND COMMUNICATIONS SECURITY, 2023, Copenhagen. **Proceedings** [...]. New York: ACM, 2023. p. 2785-2799.

SALTZER, Jerome H.; SCHROEDER, Michael D. The protection of information in computer systems. **Proceedings of the IEEE**, New York, v. 63, n. 9, p. 1278-1308, set. 1975.

SENARATH, Awanthika; ARACHCHILAGE, Nalin Asanka Gamagedara. Why developers cannot embed privacy into software systems? An empirical investigation. In: INTERNATIONAL CONFERENCE ON EVALUATION AND ASSESSMENT IN SOFTWARE ENGINEERING, 22., 2018, Christchurch. **Proceedings** [...]. New York: ACM, 2018. p. 211-216.

SPIEKERMANN, Sarah; CRANOR, Lorrie Faith. Engineering privacy. **IEEE Transactions on Software Engineering**, Los Alamitos, v. 35, n. 1, p. 67-82, 2009.

STACK OVERFLOW. **2025 Developer Survey**. [S. l.]: Stack Overflow, 2025. Disponível em: https://survey.stackoverflow.co/2025/.

SWEENEY, Latanya. K-anonymity: a model for protecting privacy. **International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems**, Singapore, v. 10, n. 5, p. 557-570, 2002.

UNIÃO EUROPEIA. Regulamento (UE) 2016/679 do Parlamento Europeu e do Conselho, de 27 de abril de 2016, relativo à proteção das pessoas singulares no que diz respeito ao tratamento de dados pessoais e à livre circulação desses dados. **Jornal Oficial da União Europeia**, Luxemburgo, L 119, p. 1-88, 4 maio 2016.

VERACODE. **2025 GenAI code security report**. Burlington: Veracode, 2025.

WOHLIN, Claes. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In: INTERNATIONAL CONFERENCE ON EVALUATION AND ASSESSMENT IN SOFTWARE ENGINEERING, 18., 2014, London. **Proceedings** [...]. New York: ACM, 2014. Article 38.

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.