

Plágio de software: Como provar, o que a lei protege e como agir juridicamente

Marcello Ávila Nascimento

A **proteção jurídica do software** no Brasil é tema de crescente relevância no contencioso empresarial, sendo, ao mesmo tempo, um dos campos mais mal compreendidos pelos titulares de programas de computador. Quando um ex-sócio, ex-funcionário ou concorrente lança no mercado um produto que parece suspeitamente idêntico ao seu, a primeira pergunta é inevitável: *isso é plágio de software, e é possível provar?* A resposta exige compreender precisamente o que a Lei 9.609/98 protege, o que ela deliberadamente deixa desprotegido, e qual é o caminho técnico-jurídico para construir uma prova robusta. Este artigo aborda cada um desses pontos com base na doutrina nacional e estrangeira mais qualificada e na jurisprudência mais recente dos tribunais superiores brasileiros.

1. O Que a Lei Protege e o Que Ela Não Protege

A Lei 9.609/98 assimila o programa de computador ao regime da obra literária (Lei 9.610/98), protegendo a **expressão** (o código na sua literalidade), não a ideia, a funcionalidade ou o resultado produzido. Conforme magistério de **Newton Silveira** em sua obra clássica sobre direito autoral de software, a dicotomia ideia-expressão é o eixo sobre o qual orbita toda a matéria: protege-se *como* o programador disse algo, não o *quê* ele disse.

Denis Borges Barbosa aprofunda essa análise ao distinguir os diferentes níveis de abstração do software (da lógica do negócio ao código-objeto), concluindo que a proteção autoral incide sobre a expressão original em cada nível, mas nunca sobre o algoritmo subjacente enquanto regra matemática ou método de operação.

No plano norte-americano, **Pamela Samuelson** (UC Berkeley) e **David Nimmer** (Nimmer on Copyright) desenvolveram o chamado *AFC test* (Abstraction, Filtration, Comparison), metodologia que filtra do corpus a ser

comparado tudo aquilo que não é protegível: elementos ditados pela eficiência, requerimentos externos (padrões de mercado, APIs), e domínio público. O que sobra, depois dessa filtragem, é o núcleo comparável. O teste é amplamente referenciado em laudos periciais brasileiros de alta complexidade.

Manoel Joaquim Pereira dos Santos e **Marcos Wachowicz** sublinham, no âmbito nacional, que a **proteção jurídica do software** abrange o código-fonte e o código-objeto na mesma medida, isto é, o executável compilado goza de proteção equivalente ao texto original, o que tem implicações diretas para as hipóteses em que o adversário não entrega voluntariamente o código-fonte.

O que *não* é protegido

Elemento	Protegido?	Fundamento
Código-fonte original	Sim	Art. 2º, Lei 9.609/98
Código-objeto compilado	Sim	Art. 2º, §1º, Lei 9.609/98
Algoritmo matemático subjacente	Não	Ideia não protegida
Funcionalidade / o que o programa faz	Não	Art. 8º, I, Lei 9.610/98
Interface gráfica (UI/UX)	Controvertido	Ver jurisprudência abaixo
Semelhança por características funcionais	Não	Art. 6º, III, Lei 9.609/98

Este último ponto (art. 6º, III) é frequentemente mal utilizado como escudo por contrafatores. O dispositivo afasta a infração quando a semelhança decorre das *características funcionais da aplicação*. Mas, como demonstrou o TJPR no caso **Gear Up x Langowski** (2024), há um limite: quando a imitação ultrapassa a semelhança funcional e alcança a experiência de interface (*look and feel*), o perito reconheceu "plágio por derivação" mesmo com códigos-fonte distintos. O STJ concedeu efeito suspensivo ao recurso especial, reconhecendo tensão com o art. 6º, III, o que sinaliza que a jurisprudência superior ainda não pacificou o tema.

2. Bibliotecas, Código Aberto e Licenças

Um ponto frequentemente ignorado pelos clientes e até por advogados menos especializados é que a presença de componentes de código aberto no software objeto da disputa afeta diretamente a análise pericial e a extensão da proteção.

Bibliotecas de terceiros (open-source, domínio público ou licenciadas) devem ser excluídas do núcleo protegível antes de qualquer comparação. O perito que não realiza essa filtragem produz um laudo tecnicamente comprometido e facilmente atacável pela parte adversa.

As licenças de software livre variam enormemente quanto à permissão de derivações:

- **MIT / BSD / Apache 2.0:** permissivas, ou seja, o código pode ser incorporado a software proprietário sem obrigação de abertura do código derivado. A cópia é permitida desde que mantidos os avisos de autoria.
- **GPL v2 / v3 (*copyleft*):** o software derivado deve ser distribuído sob a mesma licença GPL; qualquer incorporação de código GPL em produto proprietário sem observância desta regra constitui violação contratual e de direito autoral.
- **LGPL:** versão menos restritiva do GPL, admite uso em software proprietário com restrições menores.
- **Creative Commons:** aplica-se a conteúdo, não a código, sendo sua invocação para software tecnicamente incorreta.

O advogado que assessora um cliente em caso de plágio de software deve, portanto, *antes de qualquer notificação*, mapear quais bibliotecas de terceiros foram utilizadas, sob quais licenças, e se o adversário as incorporou adequadamente. Descobrir que o código "plagiado" é, na verdade, uma biblioteca GPL pública enfraquece dramaticamente a tese autoral, embora possa revelar, por outro lado, uma violação de licença igualmente relevante.

3. Como Provar o Plágio

O maior obstáculo prático em todo caso de plágio de software é o acesso ao código-fonte do concorrente. Há quatro caminhos jurídicos, em ordem crescente de complexidade:

I. Entrega voluntária: ocorre quando há relação contratual anterior (contrato de trabalho, NDA, sociedade) que produza obrigação de confidencialidade. A existência do contrato fundamenta pedido de exibição (arts. 396-404, CPC).

II. Ação de exibição de documentos: cabível quando o detentor tem obrigação legal ou contratual de exibir o código. Requer indicação precisa do documento e demonstração do interesse jurídico.

III. Tutela de urgência para preservação de prova (art. 301, CPC): indicada quando há risco de que o código seja apagado, sobrescrito ou ofuscado antes da perícia. Permite busca e apreensão, acesso a servidores e depósito em juízo.

O caso **PARADIGMA × IBID** (STJ, REsp 2.228.760-SC, rel. Min. Humberto Martins, 2026) demonstra a eficácia dessa via: a busca e apreensão dos servidores da ré, realizada sob contraditório com apresentação de quesitos pelo assistente técnico, produziu o laudo que sustentou a condenação em todas as instâncias.

IV. Engenharia reversa do binário: quando nenhum caminho anterior é viável, ferramentas como **Ghidra** (NSA, gratuita) e **IDA Pro** permitem descompilar executáveis e recuperar aproximações do código-fonte. O perito deve deixar explícito no laudo que trabalha sobre uma representação aproximada, uma vez que essa representação tem caráter indiciário, não conclusivo.

4. As Ferramentas Periciais e a Interpretação dos Resultados

As ferramentas de detecção mais utilizadas em laudos periciais brasileiros são:

- **MOSS** (Stanford): gratuito, referência acadêmica e forense, envia os arquivos para servidor externo e retorna percentual de similaridade com trechos correspondentes destacados.
- **JPlag**: open source, execução local (indicado quando há sigilo), interface HTML com visualização gráfica.
- **Plagius**: usado pelo Instituto de Criminalística de São Paulo no caso **UNITAU** (TJSP, 1005083-16.2021.8.26.0625, 2025), que apontou similaridade de apenas 0,39% entre os TCCs confrontados, resultado determinante para a improcedência.

- **CopySpider**: ferramenta antiplágio utilizada preventivamente pela própria universidade-ré no mesmo caso, que também não detectou cópia.

O percentual de similaridade isolado é insuficiente para concluir pela cópia. O perito deve identificar a *natureza* dos trechos correspondentes. **Bugs idênticos** são o melhor indício disponível: se o código do réu reproduz um defeito específico e idiossincrático do código do autor, a probabilidade de desenvolvimento independente é estatisticamente desprezível. No caso **CONSIAFI x CONNECTOR** (TJRJ, 0009398-38.2011.8.19.0209, j. 22/05/2025), o laudo pericial apontou similaridades entre o software FLINK2.jo do autor e o CONNECTOR do réu, concluindo que "é bastante plausível que o réu utilizou o software do autor para desenvolver seu próprio software", fundamento suficiente para a condenação em danos materiais, lucros cessantes e R\$ 100.000,00 de danos morais.

5. A Notificação Extrajudicial como Instrumento Legítimo

A **proteção jurídica do software** não se exerce apenas em juízo. O envio de notificações extrajudiciais aos clientes do contrafator, com exigência de regularização de licenças, foi objeto de contestação no caso **IBID x PARADIGMA** (STJ, AREsp 2.704.279-SC, rel. Min. Raul Araújo, 2026): a empresa réu buscou indenização por dano moral alegando que as notificações eram abusivas. O STJ, confirmando o TJSC, reconheceu que o envio de notificações extrajudiciais *antes mesmo do trânsito em julgado* constitui **exercício regular de direito** (art. 188, I, CC), desde que o conteúdo seja verídico e não excessivo. A decisão tem enorme relevância prática: o titular pode agir extrajudicialmente para proteger seu software sem aguardar o desfecho judicial da ação principal.

6. Estrutura do Laudo Pericial

O laudo deve traduzir achados técnicos em linguagem juridicamente operável. A estrutura recomendada contempla cinco seções: (i) identificação do perito e cadeia de custódia dos arquivos analisados (com hash SHA-256); (ii) metodologia e ferramentas utilizadas; (iii) resultados quantitativos (tabelas de percentual de similaridade por módulo); (iv) análise qualitativa, a saber: natureza dos trechos, bugs idênticos, nomes idiossincráticos de variáveis e metadados; e (v) conclusão técnica formulada em termos probabilísticos.

Formulações adequadas incluem: *"Os elementos identificados são compatíveis com reprodução direta do código do requerente, sendo a hipótese de desenvolvimento independente de baixa plausibilidade técnica"*, evitando conclusões categóricas que o perito não pode tecnicamente sustentar.

A ausência de prova pericial pode ser fatal. No caso **UNITAU**, a autora não recolheu os honorários periciais fixados em R\$ 12.544,00, e a produção da prova foi declarada preclusa. Sem a perícia, e diante dos laudos apresentados pelas rés apontando similaridade de 0,39%, o pedido foi julgado improcedente. O caso ilustra com precisão o que **Pamela Samuelson** chama de *asymmetry of proof burden*, expressão que designa o fenômeno pelo qual o ônus da prova em casos de plágio recai integralmente sobre o autor, que deve estar preparado para financiá-la.

7. Sanções Civis

A **proteção jurídica do software** no plano indenizatório opera em camadas cumulativas:

- **Art. 102, Lei 9.610/98**: apreensão dos exemplares e/ou suspensão da divulgação, sem prejuízo da indenização.
- **Art. 103, caput, Lei 9.610/98**: perdimento dos exemplares apreendidos + pagamento do preço dos vendidos, sanção civil de caráter punitivo.
- **Perdas e danos** (Código Civil): ressarcimento integral dos prejuízos demonstrados.
- **Lucros cessantes**: apurados em liquidação de sentença com base nos contratos obtidos indevidamente pelo contrafator.
- **Danos morais**: reconhecidos quando o plágio causa irresignação, aflição e abalo à honra do titular, conforme fixado no caso CONSIAFI.

No caso **PARADIGMA × IBID** (STJ, REsp 2.228.760-SC), o STJ manteve a condenação ao pagamento das licenças utilizadas indevidamente acrescida de 30% a título de sanção civil do art. 103, sem que isso configurasse *bis in idem*, desde que a cumulação respeite os princípios da razoabilidade e proporcionalidade.

Desenvolvedor, o Código que Você Escreveu Provavelmente não é seu

Antes de qualquer análise sobre plágio, há uma questão anterior que frustra boa parte dos desenvolvedores que chegam a um escritório de propriedade intelectual cheios de razão, mas saem sem caso.

A regra geral do art. 4º da Lei 9.609/98 é direta:

"Salvo estipulação em contrário, pertencerão exclusivamente ao empregador, contratante de serviços ou órgão público, os direitos relativos ao programa de computador desenvolvido durante a vigência de contrato ou vínculo estatutário, quando a atividade do empregado ou prestador de serviços inclua pesquisa e desenvolvimento."

Em termos práticos: se você foi contratado para desenvolver software (seja como CLT, PJ ou freela com contrato), o que você produziu **pertence a quem pagou**, salvo cláusula expressa em contrário. Não importa se você trabalhou 14 horas por dia, teve a ideia original, escreveu 90% do código ou o projeto foi inteiramente sua iniciativa criativa. A titularidade é do contratante.

Situações que geram confusão

1. Desenvolvido no horário de trabalho, com recursos da empresa Pertence ao empregador. Sem discussão.

2. Desenvolvido fora do horário, mas relacionado à atividade contratada Pertence ao empregador, pois a lei não exige que o desenvolvimento ocorra nas dependências da empresa ou no horário comercial. O critério é a *relação com o objeto do contrato*, não o local ou horário.

3. Desenvolvido fora do horário, com recursos próprios, sem relação com a atividade contratada. Aqui está a exceção. Se o software é genuinamente alheio ao escopo do contrato e desenvolvido sem nenhum recurso da empresa (equipamento, infraestrutura, informações privilegiadas), os direitos pertencem ao desenvolvedor, mas o §2º do art. 4º prevê que o empregador terá direito de preferência para licenciá-lo.

O que fazer antes de assinar o contrato

- Leia a cláusula de cessão de direitos intelectuais, que normalmente está nos anexos do contrato de trabalho ou de prestação de serviços.
- Se tiver projetos pessoais em andamento, **liste-os expressamente no contrato** como exceção à cessão automática.
- Documente com data certa (registro em cartório, carimbo de tempo digital, repositório com histórico auditável) os projetos anteriores ao contrato.

O que *não* adianta alegar depois

Que você teve a ideia. Que você trabalhou mais que os colegas. Que a empresa não teria chegado lá sem você. Que o contrato "não previa exatamente aquilo". A jurisprudência é consistente: na dúvida sobre o escopo do contrato, o benefício vai para o contratante, e cabe ao desenvolvedor o ônus de demonstrar que o software estava fora do objeto contratado.

Uma palavra final para o profissional de TI: o caminho não é a ação de violação de direitos autorais, mas sim a negociação prévia, antes de escrever a primeira linha de código. Depois que o software existe e foi entregue, as opções jurídicas são muito mais estreitas do que parecem.

Considerações finais

O plágio de software é um ilícito de difícil prova, mas de consequências patrimoniais expressivas para o contrafator condenado. O caminho começa muito antes do ajuizamento: identificar os contratos que justificam o acesso ao código, registrar o software no INPI para constituir data certa de autoria, mapear as licenças de componentes de terceiros e avaliar a viabilidade de uma análise pericial preliminar com MOSS ou JPlag. Se os indícios forem consistentes (especialmente a presença de bugs idênticos ou elementos idiossincráticos compartilhados), a tutela de urgência para preservação de prova é o passo seguinte.

Referências:

BRASIL. Constituição da República Federativa do Brasil de 1988. Brasília, DF: Presidência da República, 1988. Disponível em:

https://www.planalto.gov.br/ccivil_03/constituicao/constituicao.htm Acesso em: 25 maio 2026.

BRASIL. Lei n. 9.279, de 14 de maio de 1996. Regula direitos e obrigações relativos à propriedade industrial. Diário Oficial da União, Brasília, DF, 15 maio 1996. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/l9279.htm (Lei da Propriedade Industrial, incluindo disposições sobre concorrência desleal aplicáveis em litígios de software.) Acesso em: 25 maio 2026.

BRASIL. Lei n. 9.609, de 19 de fevereiro de 1998. Dispõe sobre a proteção da propriedade intelectual de programa de computador, sua comercialização no País, e dá outras providências. Diário Oficial da União, Brasília, DF, 20 fev. 1998. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/l9609.htm (Lei do Software: fundamento central do artigo, em especial arts. 2º, 4º e 6º.) Acesso em: 25 maio 2026.

BRASIL. Lei n. 9.610, de 19 de fevereiro de 1998. Altera, atualiza e consolida a legislação sobre direitos autorais e dá outras providências. Diário Oficial da União, Brasília, DF, 20 fev. 1998. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/l9610.htm (Lei de Direitos Autorais: regime subsidiariamente aplicável ao software, em especial arts. 8º, 102 e 103.) Acesso em: 25 maio 2026.

BRASIL. Lei n. 10.406, de 10 de janeiro de 2002. Institui o Código Civil. Diário Oficial da União, Brasília, DF, 11 jan. 2002. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/2002/l10406.htm (Código Civil: arts. 186, 187, 188 e 927, aplicáveis à responsabilidade civil por plágio de software.) Acesso em: 25 maio 2026.

BRASIL. Lei n. 13.105, de 16 de março de 2015. Código de Processo Civil. Diário Oficial da União, Brasília, DF, 17 mar. 2015. Disponível em: https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2015/lei/l13105.htm (CPC/2015: arts. 300, 301, 369, 381 e 396-404, referentes a tutela de urgência, produção antecipada de provas e exibição de documentos.) Acesso em: 25 maio 2026.

RIO DE JANEIRO (Estado). Tribunal de Justiça. Sentença proferida no Processo n. 0009398-38.2011.8.19.0209. 5ª Vara Cível Regional da Barra da Tijuca. Autora: Ronaldo Cesar Costa Machado Chaves. Réus: Prisma Sys Informática Comércio e Representações Ltda e Ricardo Seroa da Motta. Juíza: Adriana Angeli de Araujo de Azevedo Maia. Rio de Janeiro, RJ, 22 maio 2025. Disponível em: <https://www4.tjrj.jus.br/consultaProcessoWebV2/consultaMovimentacao.do?numProcesso=0009398-38.2011.8.19.0209> Acesso em: 25 maio 2026.

SÃO PAULO (Estado). Tribunal de Justiça. Vara da Fazenda Pública de Taubaté. Sentença proferida no Processo Digital n. 1005083-

16.2021.8.26.0625. Requerente: Juliane Carolina Anacleto Só. Requerida: Universidade de Taubaté (UNITAU) e outros. Juíza: Marcia Beringhs Domingues de Castro. Taubaté, SP, 31 out. 2025. Disponível em: <https://esaj.tjsp.jus.br/cjsg/consultaCompleta.do> Acesso em: 25 maio 2026.

BRASIL. Superior Tribunal de Justiça. Recurso Especial n. 2.228.760-SC (2024/0286132-2). Recorrentes: Paradigma Business Solutions Ltda. e IBID Sistemas de Informática Ltda. Relator: Ministro Humberto Martins. Brasília, DF, 9 mar. 2026. Publicado no DJEN/CNJ em 11 mar. 2026. Disponível em: <https://processo.stj.jus.br/SCON/> Acesso em: 25 maio 2026.

BRASIL. Superior Tribunal de Justiça. Agravo em Recurso Especial n. 2.704.279-SC (2024/0271261-9). Agravantes: IBID Sistemas de Informática Ltda. e Fabio Hoinaski. Agravado: Paradigma Business Solutions Ltda. Relator: Ministro Raul Araújo. Brasília, DF, 24 mar. 2026. Publicado no DJEN/CNJ em 26 mar. 2026. Disponível em: <https://processo.stj.jus.br/SCON/> Acesso em: 25 maio 2026.

BRASIL. Superior Tribunal de Justiça. Recurso Extraordinário nos Embargos de Declaração no Agravo Interno no Agravo em Recurso Especial n. 1.802.577-DF (2020/0324652-3). Recorrente: Factus Assessoria Empresarial Cobrança e Serviços Ltda. Recorrido: Magister Cobrança e Serviços Ltda. Relator: Ministro Jorge Mussi (Vice-Presidente). Brasília, DF, 1 jul. 2021. Publicado no DJe/STJ n. 3202 em 2 ago. 2021. Disponível em: <https://processo.stj.jus.br/SCON/> Acesso em: 25 maio 2026.

BRASIL. Superior Tribunal de Justiça. Tutela Antecipada Antecedente n. 384-PR (2024/0366596-0). Requerentes: Gear Up Brasil Tecnologia em Informação Ltda. e outros. Requerido: Langowski Internet Ltda. Relator: Ministro Raul Araújo. Brasília, DF, 27 set. 2024. Publicado no DJe/STJ n. 3962 em 30 set. 2024. Disponível em: <https://processo.stj.jus.br/SCON/> Acesso em: 25 maio 2026.

BARBOSA, Denis Borges. Uma Introdução à Propriedade Intelectual. 2. ed. Rio de Janeiro: Lumen Juris, 2003. 962 p.

NIMMER, Melville B.; NIMMER, David. Nimmer on Copyright: A Treatise on the Law of Literary, Musical and Artistic Property, and the Protection of Ideas. Nova York: LexisNexis Matthew Bender, 1963-. Obra em volumes, atualização contínua.

SAMUELSON, Pamela. The Uneasy Case for Software Copyrights Revisited. *George Washington Law Review*, Washington, DC, v. 79, n. 6, p. 1.746-1.800, ago. 2011. Disponível em: <https://scholarship.law.berkeley.edu/facpubs/1546/> Acesso em: 25 maio 2026.

SANTOS, Manoel Joaquim Pereira dos. A Proteção Autoral de Programas de Computador. Rio de Janeiro: Lumen Juris, 2008. 302 p.

SILVEIRA, Newton. Propriedade Intelectual: Propriedade Industrial, Direito de Autor, Software, Cultivares, Nome Empresarial, Abuso de Patentes. 5. ed. rev. e atual. Barueri: Manole, 2014. 378 p.

WACHOWICZ, Marcos. Propriedade Intelectual do Software e Revolução da Tecnologia da Informação. Curitiba: Juruá, 2004. 254 p.

BRASIL. Instituto Nacional da Propriedade Industrial. Registro de Programas de Computador. Rio de Janeiro: INPI, Disponível em: <https://www.gov.br/inpi/pt-br/servicos/programas-de-computador> Acesso em: 25 maio 2026.