

Jailbreak de IA

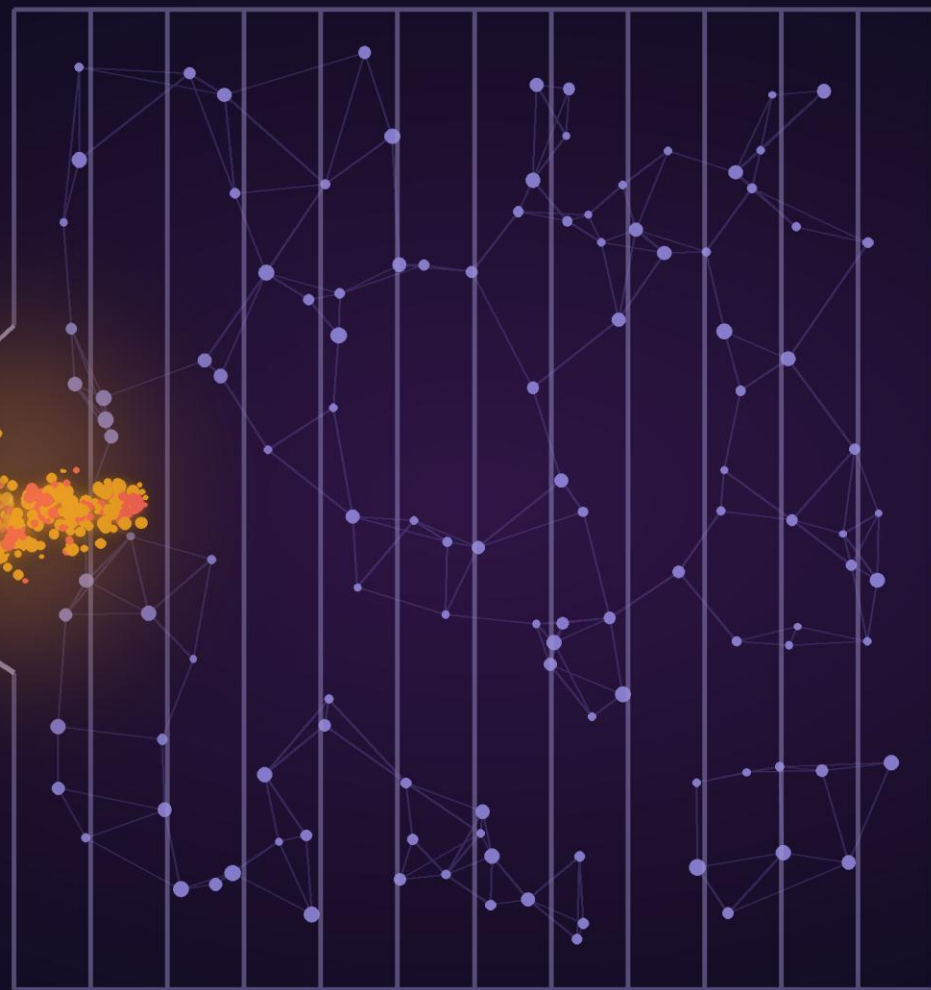
Do iPhone aos modelos de linguagem: mecanismos, ataques, defesas e governança

Aridio Silva

Pesquisador Independente

Criador do SGAEIA

Data Science e Tecnologia · GRC e Segurança da Informação



O que é jailbreak de IA

Definição

Qualquer técnica que leva um modelo de linguagem (LLM) a produzir uma resposta que seus desenvolvedores treinaram ou configuraram para recusar.

O termo nasceu em 2007, quando hackers "fugiram da prisão" do iPhone para instalar software não autorizado. Com o ChatGPT (2022), passou a designar prompts que fazem a IA ignorar suas regras de segurança.

Este estudo percorre 7 etapas: dos fundamentos à operação contínua, com foco em GRC.

1

Sem chave-mestra

No celular, o jailbreak explora falhas de código. Na IA, a arma é a própria linguagem.

2

Controle probabilístico

A segurança treinada reduz o risco de resposta nociva, mas não o elimina.

3

Risco nº 1

Prompt injection é o LLM01 do OWASP Top 10 para aplicações com LLM (2025).

Do iPhone à IA: a origem do jailbreak

SMARTPHONES

2007

Semanas após o lançamento, o iPhone Dev Team abre o sistema de arquivos; em agosto, George Hotz faz o 1º desbloqueio de operadora.

2008

Jay Freeman (saurik) lança o Cydia, loja alternativa de apps para aparelhos desbloqueados.

2010

A Biblioteca do Congresso dos EUA, por recomendação do Copyright Office, isenta da DMCA o jailbreak de smartphones.

2019

A falha checkm8 na bootrom de vários iPhones não pode ser corrigida por atualização de software.

INTELIGÊNCIA ARTIFICIAL

2022

Simon Willison cunha "prompt injection" (set.); após o ChatGPT (nov.), surgem os prompts "DAN".

2023

Wei et al. explicam as falhas; o GCG automatiza ataques; Greshake et al. definem a injeção indireta.

2024

Ataques escalam: many-shot, Crescendo e Best-of-N; surgem HarmBench e JailbreakBench.

2025

Defesas por classificadores e arquitetura: Constitutional Classifiers (Anthropic) e CaMeL (Google DeepMind).

1 ETAPA 1/7 · Fundamentos: por que funciona

OBJETIVO

Entender a causa raiz das falhas de segurança dos LLMs antes de atacar ou defender.

POR QUÊ

Sem a causa raiz, as defesas tratam sintomas: cada correção pontual é contornada pela próxima variação do ataque.

INSUMOS

- Arquitetura do LLM e fases de treino
- Técnicas de alinhamento: RLHF, IA Constitucional
- Estudos: Wei 2023, Qi 2025, Ardití 2024



O QUE É FEITO

- Analisar como o alinhamento se apoia no pré-treino
- Identificar os 4 mecanismos de falha
- Tratar a segurança como controle probabilístico



SAÍDAS

- 4 hipóteses de falha: objetivos concorrentes, generalização incompleta, alinhamento raso, recusa como direção única

IMPORTÂNCIA É a base do estudo: o conhecimento perigoso continua no modelo, e a segurança treinada reduz o risco, mas não o elimina.

2

ETAPA 2/7 · Taxonomia dos ataques

OBJETIVO

Organizar os ataques conhecidos por nível de acesso do atacante e por vetor de ataque.

POR QUÊ

Não se defende o que não se conhece: a taxonomia orienta a modelagem de ameaças e a priorização dos controles.

INSUMOS

- Mecanismos de falha (Etapa 1)
- Catálogos de prompts reais (Shen et al., 2024)
- Artigos: GCG, PAIR, TAP, many-shot



O QUE É FEITO

- Classificar o acesso: caixa-preta, cinza ou branca
- Agrupar os vetores em 8 famílias
- Ligar cada família ao mecanismo explorado



SAÍDAS

- Taxonomia de ataques
- Mapa acesso × vetor
- Cenários para red teaming

IMPORTÂNCIA Revela a evolução do artesanal (prompts em fóruns) ao industrial (ataques automatizados e escaláveis), o que muda o perfil do atacante.

3

ETAPA 3/7 · Prompt injection e agentes

OBJETIVO

Avaliar o risco quando a IA lê conteúdo de terceiros e age em sistemas reais.

POR QUÊ

Em agentes, a vítima deixa de ser a política do fornecedor e passa a ser o usuário e a organização: dados vazam e ações indevidas ocorrem.

INSUMOS

- Taxonomia de ataques (Etapa 2)
- Inventário de agentes, ferramentas e permissões
- Fontes externas: web, e-mail, anexos



O QUE É FEITO

- Distinguir jailbreak de injeção indireta
- Mapear fluxos de dados e de instruções
- Aplicar o teste da tríade letal



SAÍDAS

- Agentes de alto risco
- Cenários de exfiltração de dados
- Requisitos de arquitetura segura

IMPORTÂNCIA É o risco mais grave para empresas e ainda não tem solução equivalente às consultas parametrizadas da injeção de SQL.

4

ETAPA 4/7 · Defesas em camadas

OBJETIVO

Reduzir a chance de sucesso dos ataques e limitar o dano quando eles acontecem.

POR QUÊ

Nenhuma defesa isolada resolve: cada camada cobre falhas das outras (defesa em profundidade).

INSUMOS

- Riscos priorizados (Etapas 2 e 3)
- Técnicas de defesa da literatura
- Requisitos de utilidade e custo



O QUE É FEITO

- Reforçar o modelo: treino adversarial
- Filtrar entrada e saída com classificadores
- Isolar dados não confiáveis (CaMeL)
- Operar: monitorar, limitar, banir



SAÍDAS

- Arquitetura de defesa em 4 camadas
- Controles priorizados por risco
- Plano de implantação

IMPORTÂNCIA Troca a pergunta "como impedir todo ataque?" por "o que um ataque bem-sucedido consegue fazer?", e esta tem resposta.

5

ETAPA 5/7 · Avaliação e medição

OBJETIVO

Medir a robustez real do sistema com métricas comparáveis ao longo do tempo.

POR QUÊ

Defesas testadas só contra ataques fixos parecem melhores do que são, porque atacantes reais se adaptam.

INSUMOS

- Sistema com defesas (Etapa 4)
- Benchmarks: HarmBench, JailbreakBench, StrongREJECT, AgentDojo
- Equipe de red teaming



O QUE É FEITO

- Medir a ASR sob ataques fixos e adaptativos
- Medir recusas indevidas (over-refusal)
- Medir o custo: latência e computação



SAÍDAS

- Relatório de robustez
- Lacunas a corrigir (volta à Etapa 4)
- Linha de base para comparação

IMPORTÂNCIA Sem medição, segurança vira opinião. Medir equilibra proteção e utilidade e evita um modelo que recusa tudo.

6

ETAPA 6/7 · Governança e GRC

OBJETIVO

Integrar o risco de jailbreak aos programas de gestão de risco e conformidade.

POR QUÊ

Reguladores já cobram: desde agosto de 2025, o EU AI Act (art. 55) exige testes adversariais de modelos de propósito geral com risco sistêmico.

INSUMOS

- Resultados da avaliação (Etapa 5)
- OWASP LLM, MITRE ATLAS, NIST AI RMF, ISO/IEC 42001
- Appetite a risco da organização



O QUE É FEITO

- Inventariar e classificar os sistemas de IA
- Mapear riscos a controles e frameworks
- Definir donos e evidências



SAÍDAS

- Matriz de controles
- Políticas e procedimentos
- Evidências para auditoria

IMPORTÂNCIA Transforma achados técnicos em decisões de negócio rastreáveis, com dono, prazo e evidência.

7

ETAPA 7/7 · Operação contínua

OBJETIVO

Manter a proteção efetiva ao longo do tempo, diante de ataques novos.

POR QUÊ

É uma corrida armamentista: técnicas novas surgem continuamente, e a robustez de hoje não garante a de amanhã.

INSUMOS

- Logs de prompts e respostas
- Alertas e relatos (bug bounty)
- Novas publicações de ataques



O QUE É FEITO

- Monitorar padrões de abuso
- Tratar jailbreak como incidente de segurança
- Analisar a causa raiz e atualizar defesas



SAÍDAS

- Incidentes tratados e documentados
- Lições aprendidas
- Atualizações (volta às Etapas 2 e 4)

IMPORTÂNCIA Fecha o ciclo: o estudo deixa de ser um projeto pontual e vira um processo contínuo de melhoria, no espírito do PDCA.

Workflow do estudo: as 7 etapas em sequência



Técnicas de ataque: 8 famílias

1 Engenharia social

Personas fictícias, falsa autoridade e enquadramento hipotético ou literário.

Shen et al., 2024

2 Ofuscação

O pedido é escondido em codificações, cifras ou idiomas pouco representados.

Yong et al., 2023

3 Multi-turno

Escalada gradual, sem pedido nocivo explícito em nenhuma mensagem isolada.

Russinovich et al., 2025

4 Saturação de contexto

Muitos diálogos falsos de "obediência"; a eficácia cresce com o volume.

Anil et al., 2024

5 Gradiente (GCG)

Algoritmo gera sufixos sem sentido aparente, transferíveis entre modelos.

Zou et al., 2023

6 LLM atacante

Um modelo gera e refina tentativas contra o alvo, sem intervenção humana.

Chao et al., 2023; Mehrotra et al., 2024

7 Amostragem em massa

Muitas variações aleatórias do pedido até uma passar (Best-of-N).

Hughes et al., 2025

8 Multimodal

Instruções ou perturbações adversariais embutidas em imagens e áudio.

Qi et al., 2024

Técnicas de defesa: 4 camadas

1. Modelo

tornar o modelo mais resistente

Treino adversarial · Hierarquia de instruções (Wallace et al., 2024) · Circuit breakers (Zou et al., 2024) · Alinhamento profundo (Qi et al., 2025)

2. Filtragem

detectar fora do modelo

Llama Guard (Inan et al., 2023) · Constitutional Classifiers (Sharma et al., 2025) · Filtro de perplexidade (Jain et al., 2023) · SmoothLLM (Robey et al., 2025)

3. Arquitetura

limitar o dano possível

Spotlighting (Hines et al., 2024) · Dual LLM (Willison, 2023) · CaMeL (Debenedetti et al., 2025) · Menor privilégio e confirmação humana

4. Operação

detectar e responder

Monitoramento e alertas · Limitação de taxa e banimento de contas · Red teaming contínuo · Bug bounty

Matriz de controles para GRC

Risco	Controle recomendado	OWASP LLM 2025	MITRE ATLAS	NIST AI RMF	ISO/IEC 42001
Jailbreak direto	Classificadores de entrada e saída; política de uso; monitoramento de abuso	LLM01	AML.T0054	MEASURE 2.7; MANAGE	A.6.2.6
Prompt injection indireta	Separar instruções e dados (spotlighting, Dual LLM, CaMeL); sanitizar conteúdo externo	LLM01	AML.T0051.001	MAP; MEASURE 2.7	A.6.2.4
Agente com tríade letal	Menor privilégio; destinos permitidos; confirmação humana em ações sensíveis	LLM06; LLM02	AML.T0057	MAP; MANAGE	A.5.2; A.9
Vazamento do prompt de sistema	Sem segredos ou regras de autorização no prompt; controles fora do modelo	LLM07	AML.T0056	MEASURE	A.6.2.4
Saída nociva a jusante	Tratar a saída do LLM como entrada não confiável; validar e codificar	LLM05	—	MANAGE	A.6.2.4
Ataque não detectado	Logs de prompts e respostas; alertas; red teaming contínuo; resposta a incidentes	LLM01	AML.T0054	MANAGE 4.1	A.6.2.8; A.8.4

Códigos conferidos em out/2026 em OWASP 2025, MITRE ATLAS, NIST AI RMF 1.0 e ISO/IEC 42001 (via fontes secundárias).

Conclusão

Jailbreak de IA não é um defeito pontual, e sim uma propriedade estrutural dos LLMs atuais.

A segurança treinada é probabilística, e a mistura de instruções e dados no mesmo canal mantém aberto o espaço para manipulação.

1

Assumir o ataque

Projetar sistemas sabendo que algum jailbreak vai funcionar.

2

Limitar o dano

A arquitetura decide o que um ataque bem-sucedido consegue fazer.

3

Medir sempre

Robustez e utilidade, contra ataques fixos e adaptativos.

4

Governar com rigor

OWASP, ATLAS, NIST e ISO 42001 já cobrem o tema: aplique-os.

QUESTÕES EM ABERTO

Divulgação responsável de ataques

Liberação de modelos de pesos abertos

Quem define o que é nocivo

Robustez total é possível?

Glossário de termos técnicos (1/2)

Agente de IA

LLM que usa ferramentas e executa ações (ler e-mail, navegar, enviar dados) em nome do usuário.

Alinhamento

Treinamento que ensina o modelo a ser útil, honesto e a recusar pedidos nocivos.

ASR (Attack Success Rate)

Proporção de tentativas de ataque que produzem uma resposta nociva.

Bootrom

Código gravado no chip que inicia o aparelho; falhas nele não se corrigem por atualização.

Caixa-preta / caixa-branca

Ataque apenas pela conversa (preta) ou com acesso aos pesos do modelo (branca).

Classificador de segurança

Modelo auxiliar que analisa entrada e saída para bloquear conteúdo nocivo.

Defesa em profundidade

Camadas de controle independentes, para que a falha de uma não comprometa tudo.

Exploit

Técnica ou código que aproveita uma vulnerabilidade para obter acesso indevido.

IA Constitucional

Método da Anthropic que alinha o modelo a partir de um conjunto escrito de princípios.

Janela de contexto

Todo o texto que o modelo considera de uma vez: instruções, conversa e documentos.

Glossário de termos técnicos (2/2)

Jailbreak

Contornar as restrições impostas pelo fabricante ou desenvolvedor de um sistema.

LLM

Modelo de linguagem de grande porte, treinado para prever texto a partir de enormes volumes de dados.

Over-refusal

Recusa indevida: o modelo nega um pedido legítimo por excesso de cautela.

Prompt de sistema

Instruções do desenvolvedor que definem regras e comportamento do modelo na aplicação.

Prompt injection

Instruções maliciosas no texto processado; indireta quando vêm de conteúdo de terceiros.

Red teaming

Ataques simulados e autorizados para encontrar falhas antes de criminosos.

RLHF

Aprendizado por reforço com feedback humano: pessoas avaliam respostas e o modelo aprende com isso.

Root

Privilegio máximo de administrador; no Android, o equivalente ao jailbreak.

Sandbox

Isolamento que impede um app ou processo de acessar o restante do sistema.

Tríade letal

Dados privados + conteúdo não confiável + comunicação externa: risco crítico de vazamento.

Bibliografia verificada (1/2)

ORIGEM NOS SMARTPHONES

- MILLER, C. et al. iOS Hacker's Handbook. Wiley, 2012.
- LEVIN, J. *OS Internals, v. III: Security & Insecurity. Technogeeks, 2016 (2. ed. 2019).
- U.S. COPYRIGHT OFFICE. Section 1201 Rulemaking: exemption for smartphone jailbreaking. 2010.

FUNDAMENTOS

- WEI, A.; HAGHTALAB, N.; STEINHARDT, J. Jailbroken: How Does LLM Safety Training Fail? NeurIPS, 2023.
- QI, X. et al. Fine-tuning Aligned Language Models Compromises Safety. ICLR, 2024.
- QI, X. et al. Safety Alignment Should Be Made More Than Just a Few Tokens Deep. ICLR, 2025.
- ARDITI, A. et al. Refusal in Language Models Is Mediated by a Single Direction. NeurIPS, 2024.

ATAQUES

- PEREZ, F.; RIBEIRO, I. Ignore Previous Prompt. NeurIPS ML Safety Workshop, 2022.
- GRESHAKE, K. et al. Not What You've Signed Up For (indirect prompt injection). ACM AISEC, 2023.
- ZOU, A. et al. Universal and Transferable Adversarial Attacks on Aligned LLMs. arXiv, 2023.
- CHAO, P. et al. Jailbreaking Black Box LLMs in Twenty Queries (PAIR). arXiv:2310.08419, 2023.
- YONG, Z.-X. et al. Low-Resource Languages Jailbreak GPT-4. NeurIPS Workshop SoLaR, 2023.
- SHEN, X. et al. "Do Anything Now": In-The-Wild Jailbreak Prompts. ACM CCS, 2024.
- ANIL, C. et al. Many-shot Jailbreaking. NeurIPS, 2024.
- MEHROTRA, A. et al. Tree of Attacks. NeurIPS, 2024.
- RUSSINOVICH, M. et al. The Crescendo Multi-Turn LLM Jailbreak Attack. USENIX Security, 2025.
- HUGHES, J. et al. Best-of-N Jailbreaking. NeurIPS, 2025.
- QI, X. et al. Visual Adversarial Examples Jailbreak Aligned LLMs. AAAI, 2024.

Bibliografia verificada (2/2)

DEFESAS

- INAN, H. et al. Llama Guard. Meta, 2023.
- JAIN, N. et al. Baseline Defenses for Adversarial Attacks Against Aligned LLMs. arXiv, 2023.
- ROBEY, A. et al. SmoothLLM. TMLR, 2025.
- WALLACE, E. et al. The Instruction Hierarchy. OpenAI, 2024.
- HINES, K. et al. Defending Against Indirect Prompt Injection With Spotlighting. Microsoft, 2024.
- ZOU, A. et al. Improving Alignment and Robustness with Circuit Breakers. NeurIPS, 2024.
- SHARMA, M. et al. Constitutional Classifiers. Anthropic, arXiv:2501.18837, 2025.
- DEBENEDETTI, E. et al. Defeating Prompt Injections by Design (CaMeL). arXiv:2503.18813, 2025.
- WILLISON, S. The Dual LLM pattern (25 abr. 2023); The lethal trifecta for AI agents (16 jun. 2025). simonwillison.net.

AVALIAÇÃO

- MAZEIKA, M. et al. HarmBench. ICML, 2024.
- CHAO, P. et al. JailbreakBench. NeurIPS, 2024.
- SOULY, A. et al. A StrongREJECT for Empty Jailbreaks. NeurIPS, 2024.
- DEBENEDETTI, E. et al. AgentDojo. NeurIPS, 2024.
- SCHULHOFF, S. et al. Ignore This Title and HackAPrompt. EMNLP, 2023.

GOVERNANÇA

- OWASP. Top 10 for LLM Applications, versão 2025 (18 nov. 2024).
- MITRE. ATLAS: Adversarial Threat Landscape for AI Systems. atlas.mitre.org.
- NIST. AI 100-1 (AI RMF 1.0), 2023; AI 600-1, 2024; AI 100-2e2025 (mar. 2025).
- ISO/IEC 42001:2023. AI Management System.
- UNIÃO EUROPEIA. Regulamento (UE) 2024/1689 (AI Act). 2024.