Resultados e Discussão

Esta seção apresenta os dados obtidos a partir dos experimentos controlados de injeção de prompt e validação de tráfego. Avaliamos a eficácia do mecanismo de defesa proposto (Guardrail) em duas métricas principais: Segurança (Taxa de Detecção e Falsos Positivos) e Desempenho (Latência e Vazão).

Eficácia da Detecção de Ameaças

Os testes de eficácia submeteram o sistema a quatro cenários: tráfego benigno ("Prompt Seguro"), ataques de injeção de prompt ("Prompt Injection 1" e "2") e ataques de negação de serviço por exaustão de contexto ("Prompt Longo Demais").

Resumo da Eficácia de Segurança

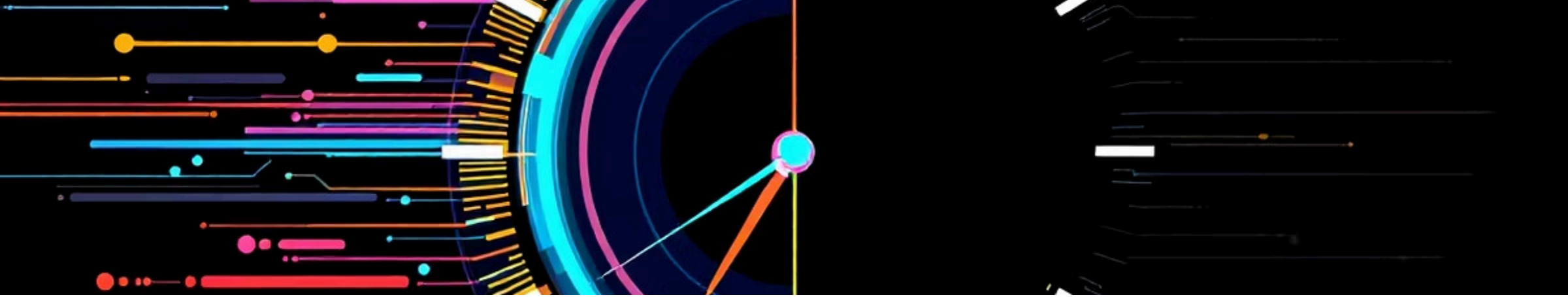
Prompt Seguro	N/A	1.0 (100%)	Bloqueio Indevido
Prompt Injection 1	1.0 (100%)	0.0	Bloqueio Correto
Prompt Injection 2	1.0 (100%)	0.0	Bloqueio Correto
Prompt Longo Demais	1.0 (100%)	0.0	Bloqueio Correto

O sistema demonstrou capacidade absoluta de defesa contra vetores de ataque conhecidos, com taxa de detecção de 1.0 para todos os ataques, sem Falsos Negativos.

Anomalia Crítica: Falsos Positivos

Observou-se uma anomalia crítica no cenário de uso legítimo: o sistema registrou uma taxa de Falsos Positivos de 1.0 para prompts seguros, classificando todo o tráfego benigno como malicioso.

Isso indica que o limiar de sensibilidade do classificador está excessivamente restritivo ou que as regras de filtragem carecem de especificidade contextual, resultando em "segurança paranoica" onde a disponibilidade do serviço é sacrificada.



Análise de Desempenho Computacional

Avaliamos o impacto da camada de segurança na latência das requisições. Os dados revelam uma assimetria interessante no tempo de processamento:

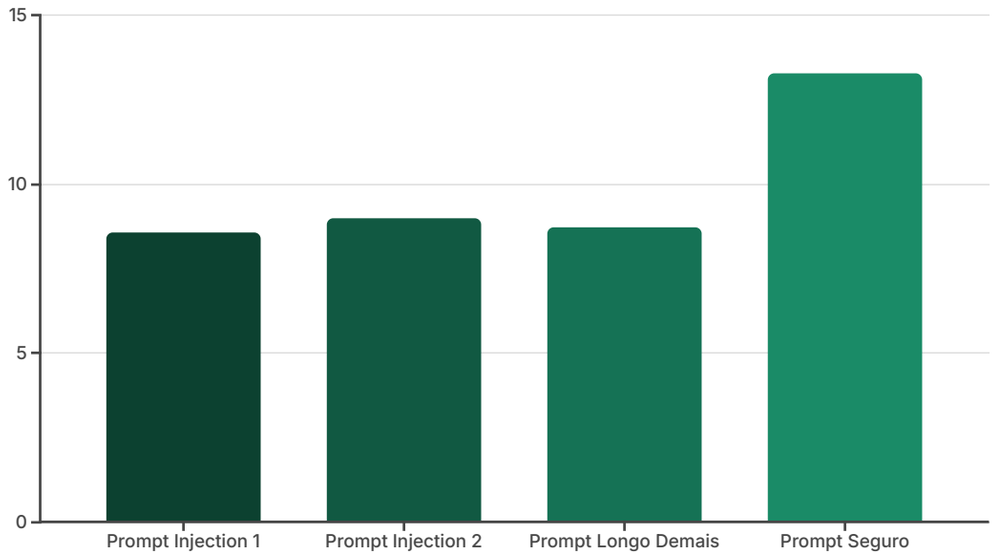
Cenários de Ataque

Latência média reduzida, entre 8.56 ms e 8.97 ms.

Cenário Benigno

Maior latência do experimento, com média de 13.28 ms.

Latência e Vazão



A vazão (*throughput*) foi significativamente maior durante os ataques (pico de 116.76 req/s) em comparação ao tráfego seguro (75.28 req/s).

Discussão: Mecanismo "Rejeição Antecipada"

A análise cruzada entre segurança e desempenho sugere que o algoritmo de defesa opera sob um mecanismo de "Rejeição Antecipada" (*Fail-Fast*).

1

Identificação Rápida

Ataques identificados nas camadas superficiais da verificação (tamanho de buffer, blacklist de tokens).

2

Interrupção Imediata

Sistema interrompe o processamento, economizando ciclos de CPU e respondendo mais rápido.

3

Otimização em Ataques

Baixa latência nos cenários de ataque devido à detecção precoce.

Custo da Validação Profunda

No caso dos "Prompts Seguros", o input parece passar por todas as camadas de verificação sem acionar os bloqueios imediatos.

Ironicamente, o sistema gasta mais recursos (tempo de processamento ~55% maior) para analisar profundamente o prompt legítimo, apenas para rejeitá-lo ao final do pipeline devido à má calibração.

Conclusão da Análise

Os resultados validam a arquitetura como altamente resiliente a ataques, mas inviável para produção no estado atual devido à taxa de 100% de falsos positivos.

Resiliência a Ataques

Sistema ideal sob ataque, bloqueando rápido e com baixo custo computacional.

Falha na Usabilidade

Falha na distinção de nuances semânticas em interações legítimas.

Próximos Passos

Trabalhos futuros devem focar no ajuste fino (*fine-tuning*) dos parâmetros de decisão para reduzir a taxa de rejeição de usuários legítimos.

01

Ajuste Fino

Calibrar parâmetros para reduzir falsos positivos.

02

Equilíbrio

Balancear segurança robusta com usabilidade necessária.

03

Implementação

Garantir que o sistema seja viável para aplicações reais.