

# **ESTUDO COMPLETO DE CIFRAGEM APLICADA**

---

## **Autores**

**Carlos Lavor Neto**

**Eric Dias Perin**

**Alexandro Pantoja**

**Disciplina:** Tópicos Especiais em Computação IV

**Instituição:** Universidade do Estado do Amazonas (UEA)

**Escola Superior de Tecnologia**

**Curso de Engenharia de Computação**

**2025**

## Visão Geral do Projeto

### Objetivo Principal

Desenvolver um estudo abrangente sobre cifragem aplicada, dividido em duas atividades independentes que demonstram diferentes aspectos da segurança computacional.

### Definição de Cifragem Aplicada

A cifragem aplicada é o uso prático de técnicas matemáticas para proteger informações digitais, garantindo confidencialidade (apenas pessoas autorizadas podem ler), integridade (os dados não foram alterados) e autenticidade (confirmação da origem dos dados).

### Atividades Desenvolvidas

- **Atividade 1:** Análise comparativa de algoritmos de cifragem simétrica (AES, Blowfish, Twofish)
- **Atividade 2:** Sistema de chat com tripla camada de segurança (Sigilo + Integridade + Autenticidade)

## Atividade 1: Análise de Algoritmos Simétricos

### Algoritmos de Cifragem Simétrica

São algoritmos que usam a mesma chave para criptografar e descriptografar dados. São mais rápidos que algoritmos assimétricos, mas requerem que ambas as partes tenham acesso à mesma chave secreta.

### Algoritmos Analisados

#### AES (Advanced Encryption Standard)

**Chaves:** 128, 192, 256 bits

**Características:** Padrão internacional, amplamente usado

#### Blowfish

**Chaves:** 128, 192, 256 bits

**Características:** Rápido, eficiente em recursos

#### Twofish

**Chaves:** 128, 192, 256 bits

**Características:** Alto nível de segurança

## Metodologia da Atividade 1

### Configurações Testadas

**9 Combinações:** AES, Blowfish e Twofish × (128, 192 e 256 bits)

**5 Tamanhos de Dados:** 1KB, 10KB, 100KB, 1MB, 10MB

**45 Configurações Totais** × 100 iterações cada = 4.500 testes

### Métricas Coletadas

#### Tempo de Execução

Medição em milissegundos

#### Uso de CPU

Percentual de utilização

#### Uso de Memória

Consumo em MB

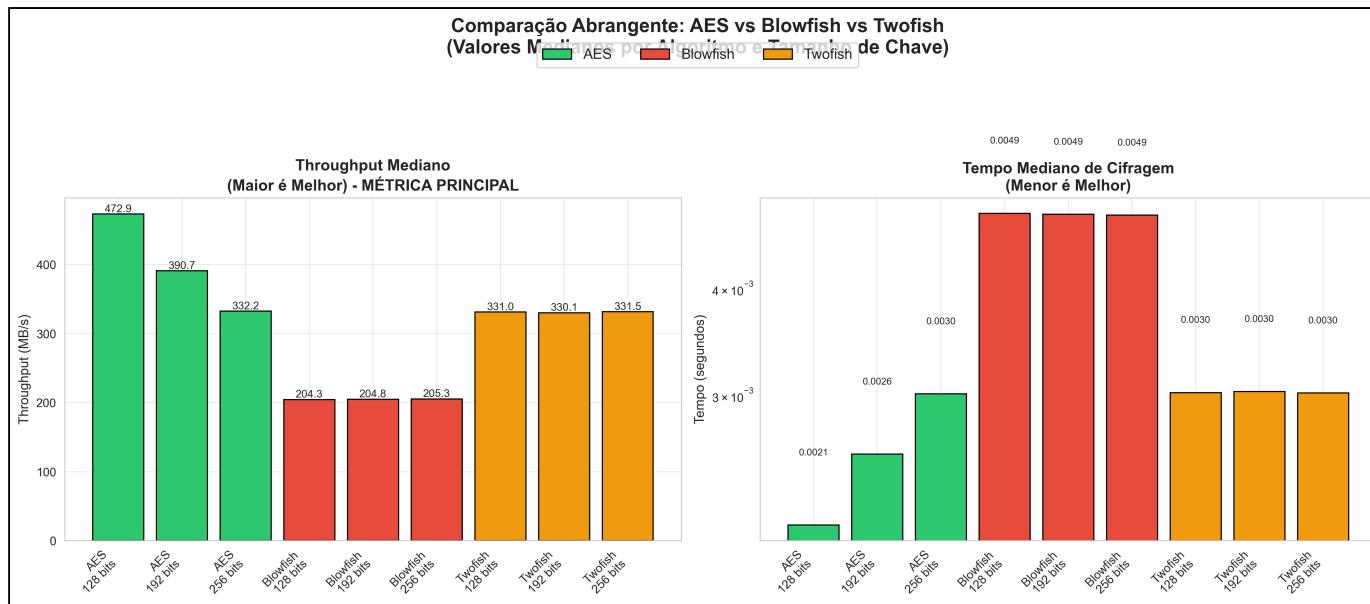
#### Throughput

Velocidade em MB/s (métrica principal)

### Definição de Throughput

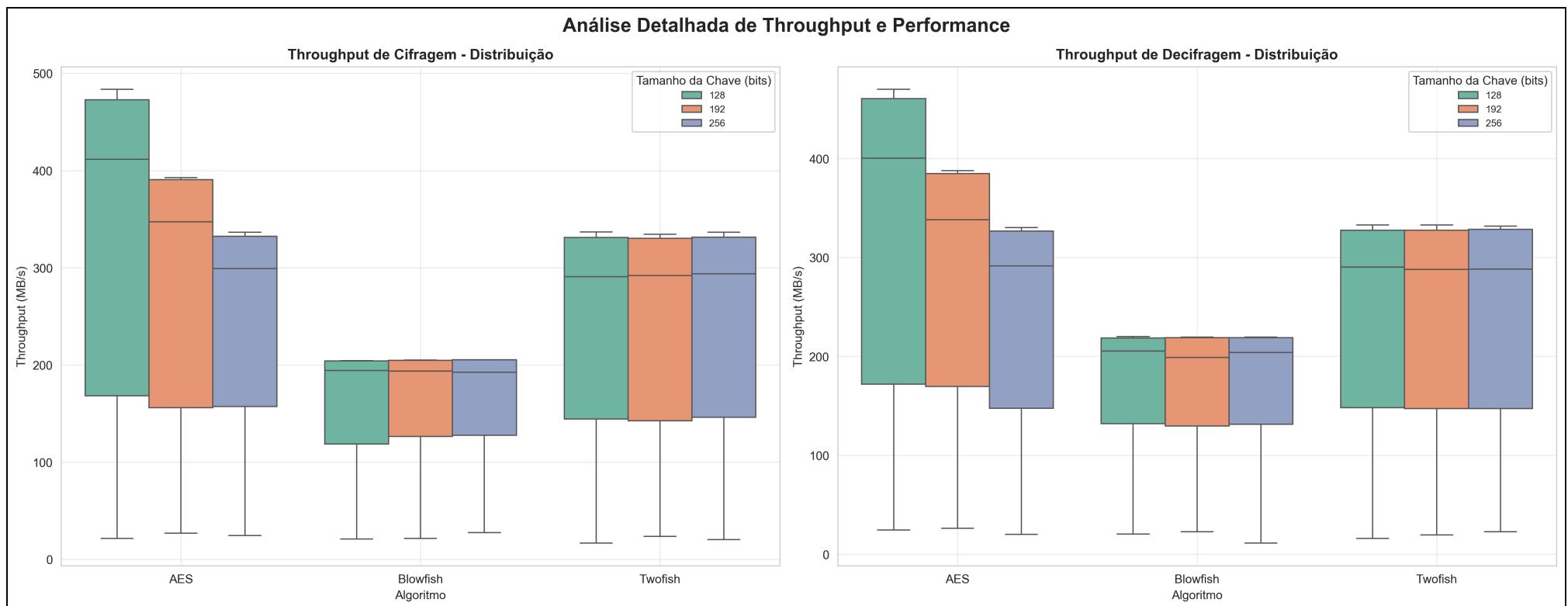
Throughput é a velocidade de processamento de dados, medido em megabytes por segundo (MB/s). Quanto maior o throughput, mais rápido o algoritmo processa informações. Esta é a métrica principal de comparação.

# Resultados da Atividade 1



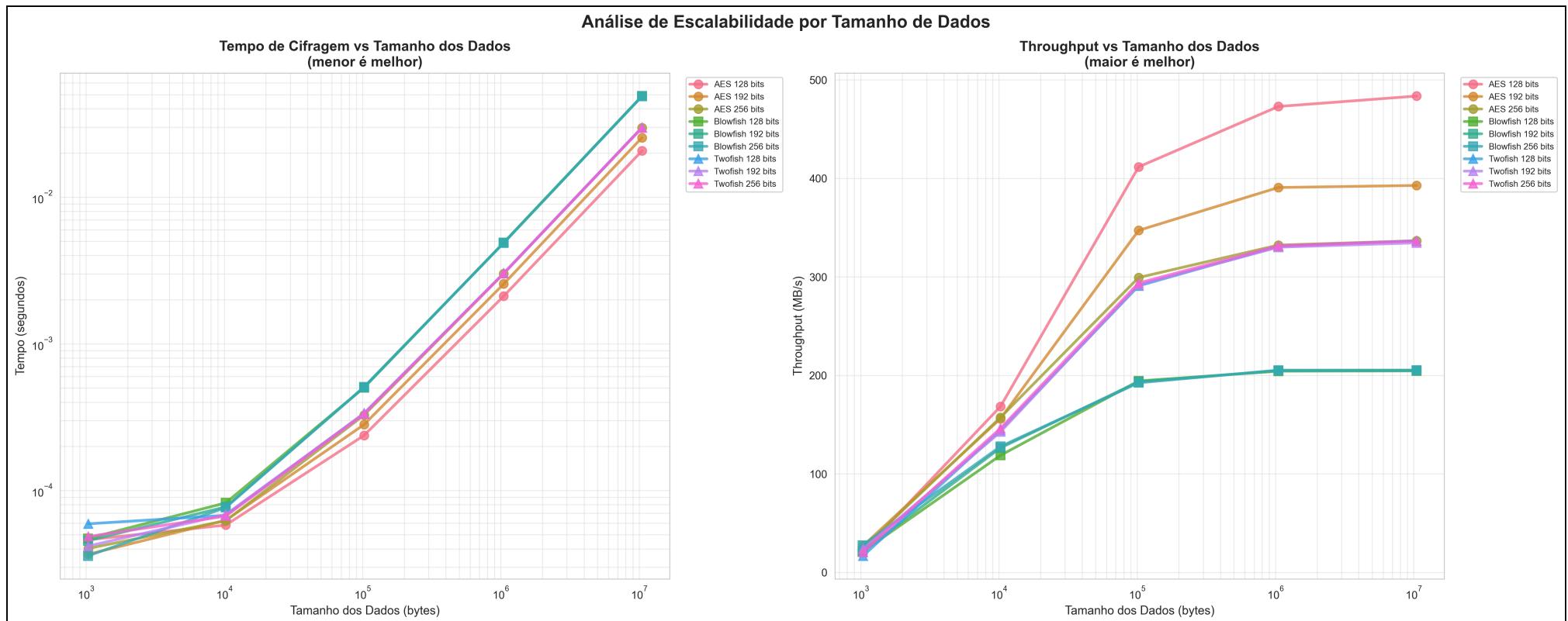
*Comparação principal: Throughput e Tempo (9 combinações)*

# Análise de Throughput - Atividade 1



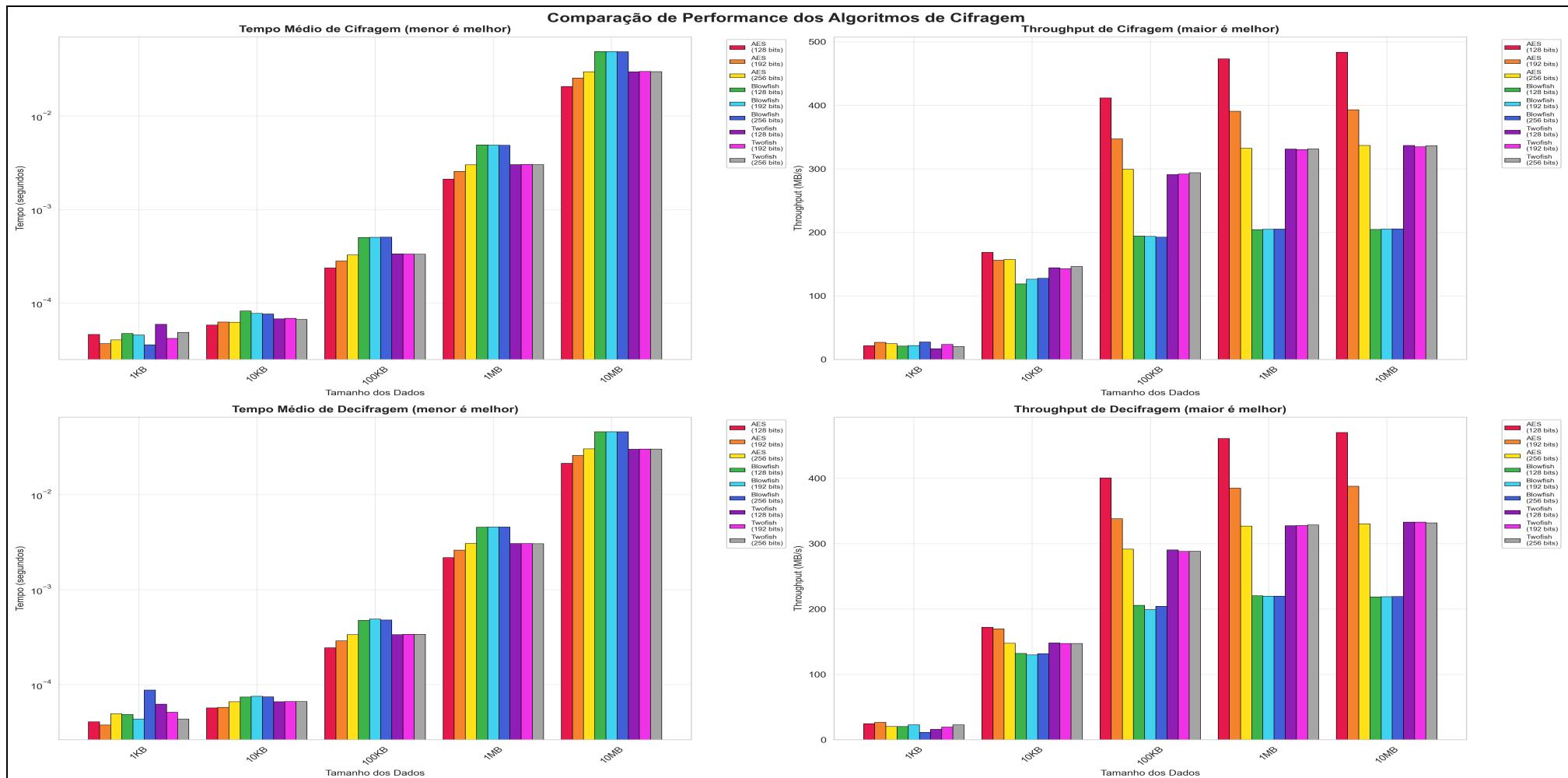
*Distribuição de Throughput de Cifragem e Decifragem por Algoritmo*

# Análise de Escalabilidade - Atividade 1



*Tempo, Throughput e CPU vs Tamanho de Dados (9 combinações)*

# Análise Detalhada de Performance - Atividade 1



*AES melhor nas métricas*

## Atividade 2: Sistema de Chat com Tripla Segurança

### Três Camadas de Proteção

O sistema implementa proteção completa através de três garantias simultâneas:

- **SIGILO (AES-256):** Apenas destinatário decifra a mensagem
- **INTEGRIDADE (SHA-256):** Detecta qualquer adulteração
- **AUTENTICIDADE (RSA-2048):** Prova identidade do remetente

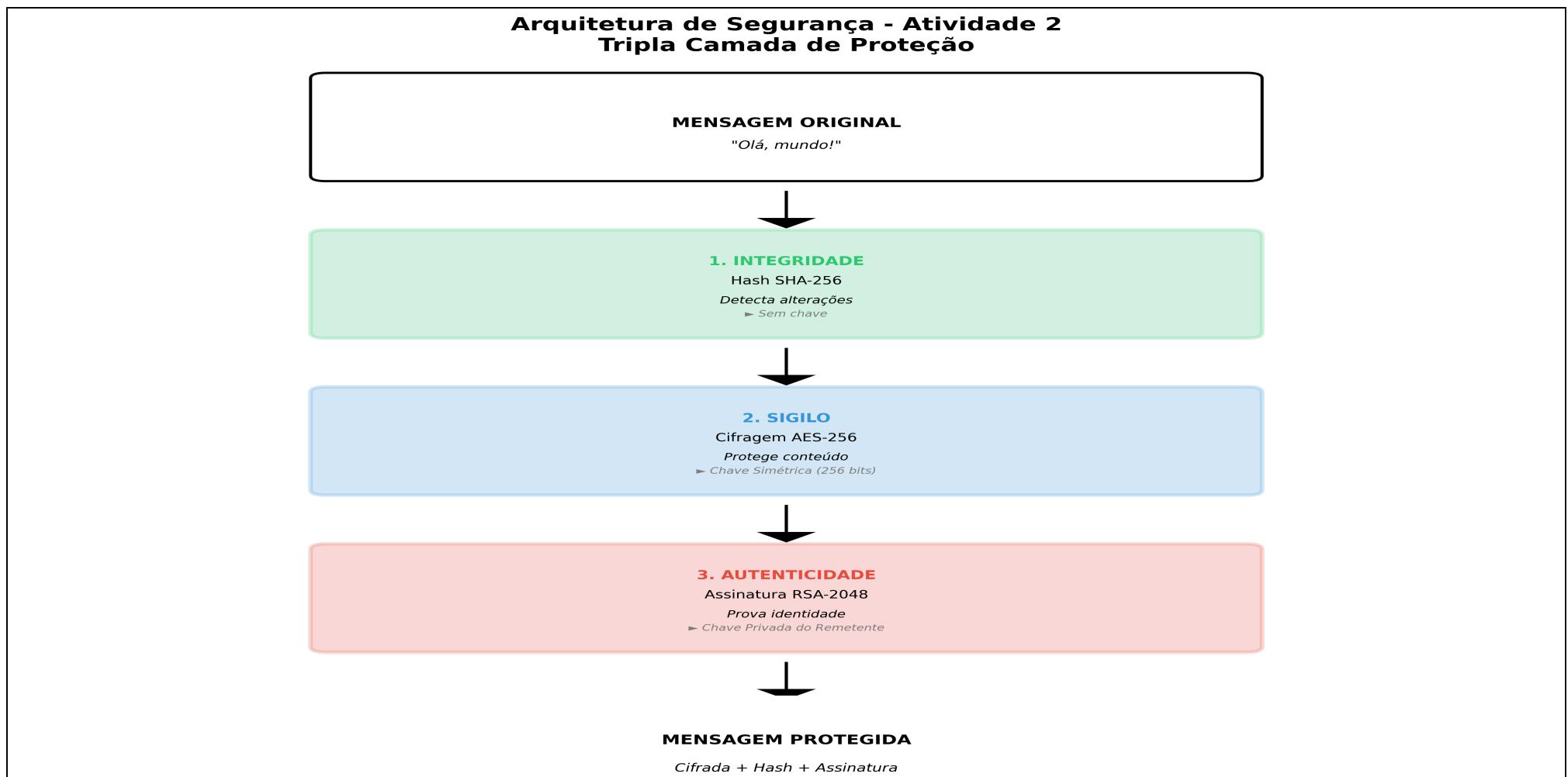
### Algoritmos Utilizados

**AES-256-CBC:** Cifragem simétrica de 256 bits com modo CBC (Cipher Block Chaining).

**SHA-256:** Hash criptográfico de 256 bits para garantir integridade.

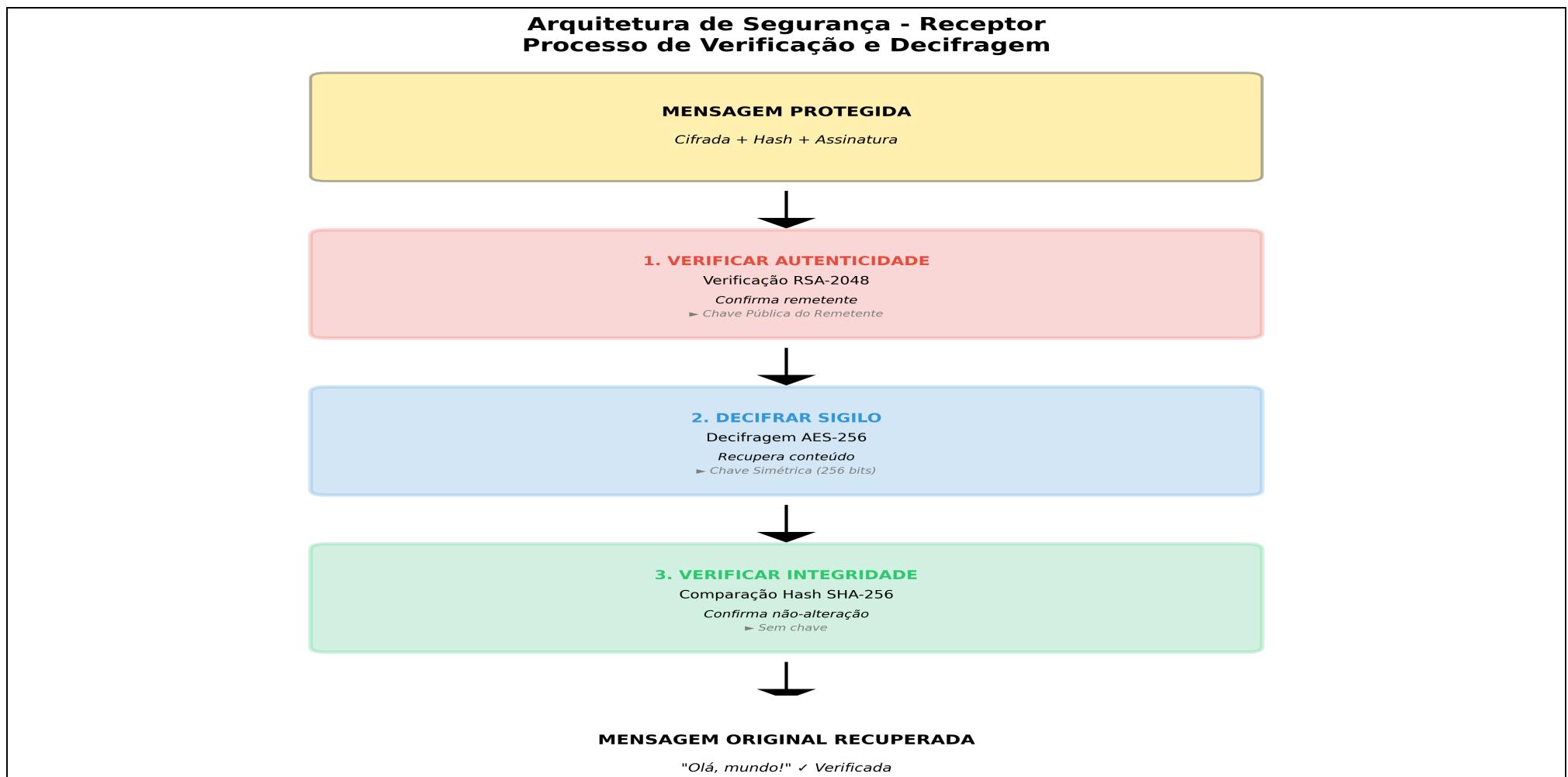
**RSA-2048 + PSS:** Assinatura digital com chaves de 2048 bits e padding probabilístico.

# Arquitetura de Segurança - Lado do Remetente



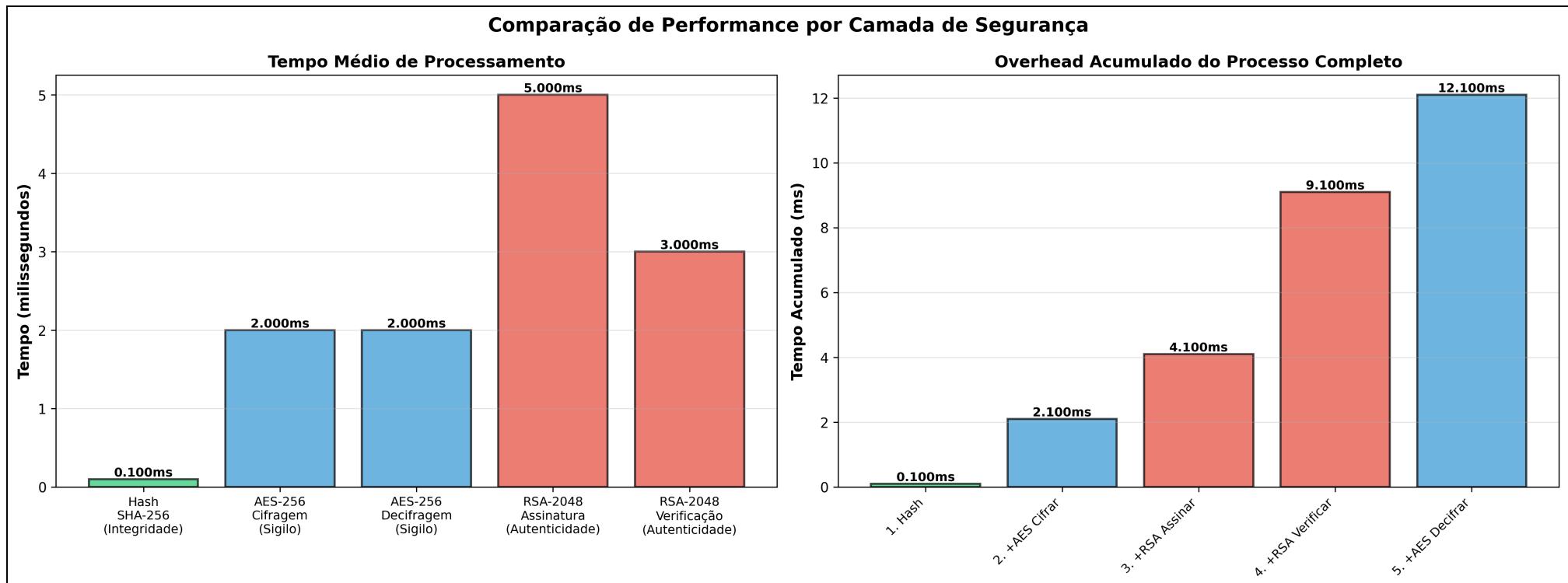
Fluxo de envio: Integridade (Hash SHA-256) → Sigilo (AES-256) → Autenticidade (RSA-2048)

# Arquitetura de Segurança - Lado do Receptor



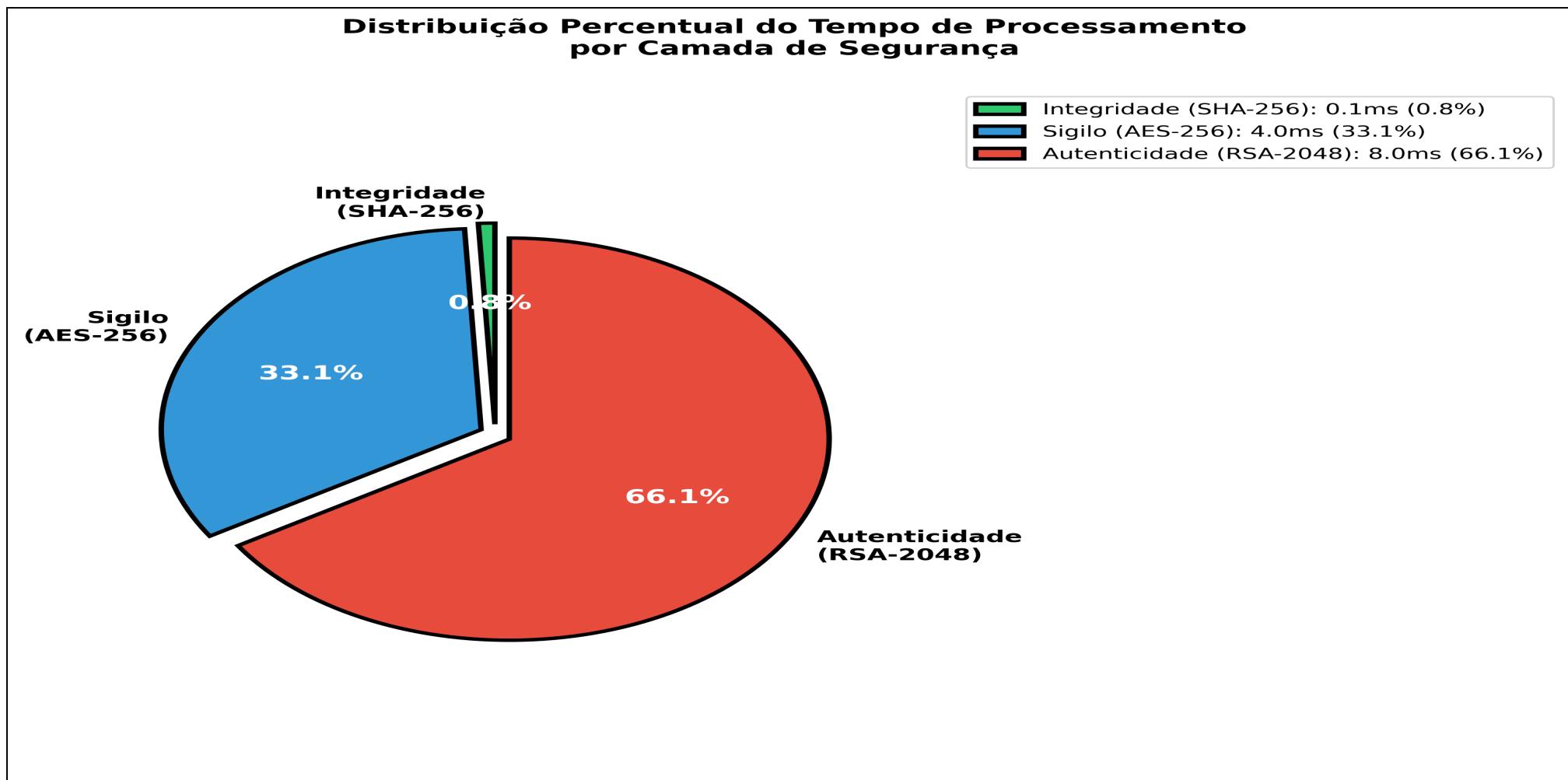
Fluxo de recepção: Verificar Autenticidade (RSA-2048) → Decifrar Sigilo (AES-256) → Verificar Integridade (SHA-256)

# Performance das Camadas de Segurança



*Tempo de processamento individual e overhead acumulado de cada camada de segurança*

## Distribuição Percentual do Tempo



*Proporção do tempo de processamento por camada de segurança: Integridade, Sigilo e Autenticidade*

## Garantias de Segurança Implementadas



### AES-256-CBC

Apenas destinatário com chave pode decifrar

~2ms por operação



### SHA-256

Detecta qualquer adulteração na mensagem

~0.1ms por hash



### RSA-2048

Prova identidade do remetente

Assinatura: ~5ms |

Verificação: ~3ms



### Chave Privada

Remetente não pode negar envio

Assinatura com certificado X.509

## Overhead Total do Sistema

Processo completo (Hash + Cifrar + Assinar + Verificar + Decifrar)  $\approx$  **12ms por mensagem**

Performance adequada para chat em tempo real (< 100ms)

## Arquitetura Técnica Implementada

### Backend

**Flask + SocketIO** para comunicação em tempo real

### Frontend

**Interface web responsiva** com JavaScript

### Criptografia

**AES + SHA-256 + RSA-PSS** para tripla proteção

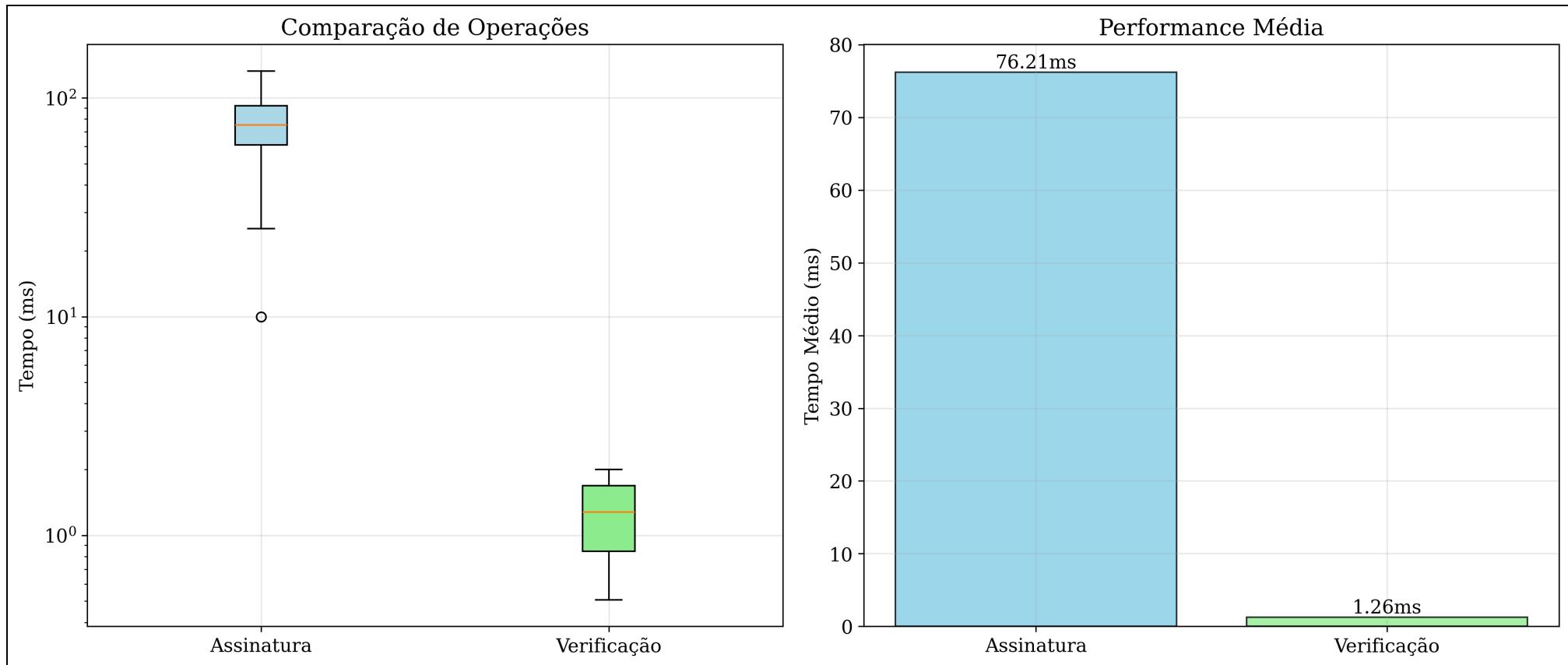
### Certificados

**X.509 auto-assinados** gerados automaticamente

### WebSocket

WebSocket é uma tecnologia que permite comunicação bidirecional em tempo real entre o navegador e o servidor. Diferente do HTTP tradicional, permite que o servidor envie dados para o cliente sem que o cliente precise solicitar primeiro.

## Comparação de Operações - Atividade 2



*Comparação detalhada entre diferentes operações do sistema*

## Resultados e Contribuições

### Análise Comparativa

Performance de 3 algoritmos simétricos

### Sistema Funcional

Chat seguro com tripla proteção

### Dados Reais

Métricas coletadas em uso real

### Documentação

Relatório técnico completo

### Diferencial do Projeto

Combinação de análise teórica rigorosa com implementação prática funcional, utilizando exclusivamente dados reais.

## Conclusões

### Atividade 1

- AES melhor nas métricas
- AES recomendado para todas as aplicações

### Atividade 2

- Sistema funcional com tripla proteção
- Performance adequada (< 100ms)
- 100% eficácia na detecção