# 📝 Controle de Incidentes

## 🎯 Resumo do Incidente

- Serviço impactado: payment-api  
- Ambiente: Produção (EKS-prod)  
- Namespace: payment  
- Data/Hora de Início: YYYY-MM-DD HH:MM  
- Data/Hora de Fim: YYYY-MM-DD HH:MM  
- Duração: XX minutos  
- Impacto: Alta latência e falhas no processamento de pagamentos. Serviço apresentou >2s de resposta por 10 minutos.

## 🧭 Linha do Tempo

|  |  |
| --- | --- |
| **Horário** | **Evento** |
| 10:05 | Alerta de latência da API acionado (> 2s por 10min) |
| 10:10 | Logs mostram timeout na comunicação com RDS |
| 10:15 | Confirmado gargalo no pool de conexões com RDS |
| 10:30 | Escalado número de réplicas + ajuste do pool |
| 10:45 | Métricas estabilizadas, latência voltou ao normal |

## 🧩 Causa Raiz

Aumento de carga inesperado levou à saturação do pool de conexões com o banco RDS.   
Como consequência, a aplicação apresentou timeouts e aumento expressivo na latência média das requisições HTTP.

## 🔥 Detecção

- Alerta automático de latência média > 2s  
- Logs com erros timeout ao acessar banco  
- Traces via OpenTelemetry mostraram atraso no acesso a serviços externos

## 🛠 Ações Tomadas

|  |  |
| --- | --- |
| **Ação** | **Status** |
| Verificação de pods e métricas de recursos | ✅ |
| Escalonamento das réplicas de payment-api | ✅ |
| Ajuste no pool de conexões da aplicação | ✅ |
| Verificação de locks/slow queries no RDS | ✅ |
| Análise de logs e traces distribuídos | ✅ |

## 🧪 Hipóteses Verificadas

- [x] Saturação do pool de conexões com RDS  
- [ ] Problemas na rede (NAT Gateway)  
- [ ] Mudança recente no código (não houve)  
- [ ] Sobrecarga de CPU/memória

## ✅ Lições Aprendidas

- A saturação de banco é silenciosa, mas crítica  
- Métricas de conexão com banco não estavam sendo monitoradas  
- Escalabilidade horizontal ajudou, mas o gargalo era o pool fixo

## 🛡 Melhorias Preventivas

|  |  |
| --- | --- |
| **Ação Preventiva** | **Status** |
| Adicionar monitoramento do uso do pool de conexões | 🔧 A fazer |
| Habilitar HPA baseado em latência | 🔧 A fazer |
| Melhorar limites de recurso dos pods | ✅ Feito |
| Implementar fallback para falha no banco | 🔧 A fazer |
| Adicionar alertas para DBConnectionTimeout | ✅ Feito |

## 📣 Comunicação

- Canal: #ops-alerts, reuniões de status a cada 15 minutos  
- Atualizações contínuas foram enviadas até a normalização  
- PIR compartilhado com time de engenharia e SRE

## 🗂 Artefatos Relacionados

- Logs e métricas no Grafana: [link]  
- Traces de requisições: [link]  
- Pull request de ajuste do pool: [link]  
- Configuração ArgoCD: [link]

Revisado por: Alexandre José Batista

Data: 14/07/2025