Plano de Segurança – Bot Cripto Telegram

# 1. Introdução

Este Plano de Segurança define práticas e controles para proteger o Bot Cripto Telegram contra acessos não autorizados, vazamento de dados e uso indevido de APIs. O objetivo é garantir confidencialidade, integridade e disponibilidade do sistema.

# 2. Autenticação e Controle de Acesso

- \*\*ChatID autorizado\*\*: apenas IDs definidos podem interagir com o bot.  
- \*\*Webhook Telegram\*\*: protegido por token secreto exclusivo.  
- \*\*N8N Vault\*\*: todas as credenciais (Binance, Supabase, OpenAI) armazenadas em cofre seguro.  
- \*\*MFA (quando disponível)\*\*: habilitar autenticação multifator nas contas vinculadas (Supabase, Upstash, etc.).

# 3. Proteção de Dados

- \*\*Supabase/Postgres\*\*: acesso restrito apenas ao N8N e operador autorizado.  
- \*\*Redis\*\*: proteger com senha/token e TLS (Upstash ou configuração própria).  
- \*\*Criptografia em trânsito\*\*: todas as comunicações devem usar HTTPS/TLS.  
- \*\*Criptografia em repouso\*\*: Supabase já provê criptografia em disco nativamente.  
- \*\*Logs sensíveis\*\*: remover/anonimizar chaves ou dados confidenciais de logs.

# 4. Segurança Operacional

- \*\*Segregação de ambientes\*\*: separar ambientes de teste e produção, com credenciais distintas.  
- \*\*Principle of Least Privilege (PoLP)\*\*: cada chave de API deve ter apenas as permissões necessárias (ex.: chave Binance read-only).  
- \*\*Rotação de credenciais\*\*: redefinir periodicamente chaves e tokens críticos.  
- \*\*Monitoramento de acessos\*\*: registrar e revisar tentativas de acesso suspeitas.

# 5. Gestão de Vulnerabilidades

- \*\*Atualizações regulares\*\*: manter N8N, dependências e banco de dados sempre atualizados.  
- \*\*Bibliotecas de IA (LangChain/MCP)\*\*: acompanhar changelogs e corrigir vulnerabilidades conhecidas.  
- \*\*Auditorias periódicas\*\*: revisar código, workflows e permissões.  
- \*\*Backup e recovery\*\*: políticas de backup do Supabase + restore testado regularmente.

# 6. Plano de Resposta a Incidentes

1. \*\*Detecção\*\*: incidente identificado via WF\_Error\_Handler, logs ou monitoramento externo.  
2. \*\*Comunicação\*\*: notificação imediata no Telegram e registro em ErrorState.  
3. \*\*Mitigação\*\*: isolar credenciais afetadas, desativar workflows comprometidos.  
4. \*\*Recuperação\*\*: restaurar dados de backup, gerar novas credenciais.  
5. \*\*Análise pós-incidente\*\*: documentar causa raiz e aplicar medidas preventivas.

# 7. Conclusão

Com este plano de segurança, o Bot Cripto Telegram adota uma abordagem proativa contra riscos digitais, assegurando que mesmo em caso de incidentes o impacto seja minimizado e a continuidade garantida.