



RENDER.COM DEPLOYMENT GUIDE - ADK Arbitrage Profit Guard



Pré-requisitos

1. **Conta no Render.com** - [Criar conta](https://render.com) (<https://render.com>)
2. **Repositório GitHub** - Fork ou clone do ADKArbitrageProfitGuard
3. **API Keys das Exchanges** - Binance, OKX, Bybit (com permissões de trading)
4. **Proxies (Opcional)** - Para contornar geo-bloqueio



Configuração de Geo-Bypass

Opção 1: Proxies Residenciais (Recomendado)

```
# Exemplos de provedores confiáveis:  
# - Bright Data (ex-Luminati)  
# - Oxylabs  
# - Smartproxy  
# - ProxyMesh  
  
# Formato das URLs:  
PROXY_URL=http://username:password@proxy-host:port  
PROXY_URL_2=socks5://username:password@socks-host:port
```

Opção 2: VPN Services

```
# Configurar através de proxy HTTP/SOCKS5  
PROXY_URL=http://vpn-proxy:port
```

Opção 3: Render Internal Proxy (Incluído)

O sistema inclui um proxy interno configurado automaticamente via `render.yaml`.



Deploy no Render.com

Passo 1: Conectar Repositório

1. Acesse [Render Dashboard](https://dashboard.render.com) (<https://dashboard.render.com>)
2. Clique em "New +" → "Web Service"
3. Conecte seu repositório GitHub
4. Selecione o repositório ADKArbitrageProfitGuard

Passo 2: Configuração Básica

```
# Configurações automáticas via render.yaml
Name: adk-arbitrage-profit-guard
Runtime: Node
Region: Oregon (recomendado para exchanges)
Branch: main
Build Command: npm ci && npm run build
Start Command: npm start
```

Passo 3: Variáveis de Ambiente

🔑 API Keys das Exchanges (OBRIGATÓRIO)

```
# Binance
BINANCE_API_KEY=your_real_binance_api_key
BINANCE_API_SECRET=your_real_binance_secret

# OKX
OKX_API_KEY=your_real_okx_api_key
OKX_API_SECRET=your_real_okx_secret
OKX_PASSPHRASE=your_real_okx_passphrase

# Bybit
BYBIT_API_KEY=your_real_bybit_api_key
BYBIT_API_SECRET=your_real_bybit_secret
```

🌐 Configuração de Proxy (OPCIONAL)

```
# Habilitar sistema de proxy
PROXY_ENABLED=true

# URLs de proxy (configure pelo menos uma)
PROXY_URL=http://username:password@proxy1.example.com:8080
PROXY_URL_2=socks5://username:password@proxy2.example.com:1080
PROXY_URL_3=http://username:password@proxy3.example.com:3128

# Proxies residenciais (recomendado)
PROXY_URL_RESIDENTIAL_1=http://user:pass@residential1.provider.com:port
PROXY_URL_RESIDENTIAL_2=http://user:pass@residential2.provider.com:port

# Proxies datacenter (backup)
PROXY_URL_DATACENTER_1=http://user:pass@datacenter1.provider.com:port
PROXY_URL_DATACENTER_2=http://user:pass@datacenter2.provider.com:port
```

🛡️ Configuração de Segurança

```
# JWT Secret (auto-gerado pelo Render)
JWT_SECRET=auto-generated-by-render

# Trading Safety
REAL_TRADING_ENABLED=false # Mude para true apenas após testes
PAPER_TRADING_MODE=true    # Mude para false para trading real
```

Configuração de Trading

```
# Pares de trading
PAIRS=BTC/USDT,ETH/USDT,BNB/USDT,SOL/USDT,ADA/USDT

# Parâmetros de arbitragem
BASIS_ENTRY=0.001
BASIS_EXIT=0.0005
MAX_NOTIONAL_USDT=1000
MAX_DAILY_TRADES=20
MIN_PROFIT_THRESHOLD=0.001

# Performance
RATE_LIMIT_REQUESTS_PER_MINUTE=100
CACHE_TTL_SECONDS=30
```

Passo 4: Deploy

1. Clique em "Create Web Service"
2. Aguarde o build e deploy (5-10 minutos)
3. Acesse a URL fornecida pelo Render

Verificação de Saúde

Health Check Endpoint

```
# Verificação rápida
curl https://your-app.onrender.com/health

# Verificação completa
curl https://your-app.onrender.com/api/health/full
```

Logs de Sistema

```
# No Render Dashboard:
# 1. Acesse seu serviço
# 2. Clique na aba "Logs"
# 3. Monitore os logs em tempo real
```

Troubleshooting

Problema: Geo-bloqueio das Exchanges

```
# Solução 1: Configurar proxies
PROXY_ENABLED=true
PROXY_URL=http://your-proxy:port

# Solução 2: Verificar logs
# Procure por mensagens como:
# "❌ Geo-blocked response: 403 Forbidden"
# "🌐 Switching to next proxy..."
```

Problema: Falha na Conexão com Exchange

```
# Verificar API keys
# Logs mostrarão:
# "❌ Invalid API key" ou "❌ Signature verification failed"

# Solução:
# 1. Verificar API keys no Render Dashboard
# 2. Confirmar permissões nas exchanges
# 3. Verificar whitelist de IPs (se aplicável)
```

Problema: Performance Lenta

```
# Solução 1: Upgrade do plano Render
# Starter → Standard → Pro

# Solução 2: Otimizar configurações
RATE_LIMIT_REQUESTS_PER_MINUTE=50 # Reduzir se necessário
CACHE_TTL_SECONDS=60               # Aumentar cache
```



Monitoramento

Métricas Importantes

- **Uptime:** Disponibilidade do serviço
- **Latência:** Tempo de resposta das APIs
- **Taxa de Sucesso:** % de trades bem-sucedidos
- **Uso de Proxy:** Eficácia do geo-bypass

Alertas Recomendados

1. **Falha de Conexão:** > 5 falhas consecutivas
2. **Geo-bloqueio:** Detecção de bloqueio regional
3. **API Rate Limit:** Excesso de requisições
4. **Memória:** Uso > 80% da RAM disponível



Segurança

Boas Práticas

1. **Nunca commitar API keys** no código
2. **Usar HTTPS** sempre (automático no Render)
3. **Monitorar logs** regularmente
4. **Testar em paper trading** antes do real
5. **Configurar alertas** para falhas críticas

Backup e Recovery

```
# Backup automático via Render
# 1. Database backups (se usando PostgreSQL)
# 2. Environment variables backup
# 3. Code backup via GitHub
```



Otimização de Performance

Configurações Recomendadas por Plano

Starter Plan (\$7/mês)

```
MAX_NOTIONAL_USDT=500
MAX_DAILY_TRADES=10
RATE_LIMIT_REQUESTS_PER_MINUTE=50
```

Standard Plan (\$25/mês)

```
MAX_NOTIONAL_USDT=2000
MAX_DAILY_TRADES=50
RATE_LIMIT_REQUESTS_PER_MINUTE=100
```

Pro Plan (\$85/mês)

```
MAX_NOTIONAL_USDT=10000
MAX_DAILY_TRADES=200
RATE_LIMIT_REQUESTS_PER_MINUTE=200
```



Suporte

Recursos de Ajuda

1. **Render Documentation:** docs.render.com (<https://docs.render.com>)
2. **Render Community:** community.render.com (<https://community.render.com>)
3. **GitHub Issues:** Para bugs específicos do ADK
4. **Health Check:** `/api/health/full` para diagnósticos

Contatos de Emergência

- **Render Support:** Via dashboard ou email
- **Exchange Support:** Para problemas de API
- **Proxy Provider:** Para problemas de conectividade



Checklist de Deploy

- ☐ Repositório conectado ao Render
- ☐ Variáveis de ambiente configuradas
- ☐ API keys das exchanges adicionadas
- ☐ Proxies configurados (se necessário)
- ☐ Health check funcionando
- ☐ Logs sendo monitorados
- ☐ Paper trading testado
- ☐ Alertas configurados
- ☐ Backup strategy definida

 Parabéns! Seu sistema ADK Arbitrage Profit Guard está pronto para produção no [Render.com](https://render.com)!