

Documentação Completa - Stack Zabbix Enterprise

Índice

1. [Visão Geral](#)
 2. [Arquitetura](#)
 3. [Estrutura de Arquivos](#)
 4. [Serviços](#)
 5. [Configuração](#)
 6. [Dashboards](#)
 7. [Manutenção](#)
 8. [Troubleshooting](#)
 9. [Backup e Restore](#)
 10. [Monitoramento](#)
-



Visão Geral

Stack completo de monitoramento enterprise com capacidade para **2.000-3.000 hosts simultâneos**.

Componentes Principais:

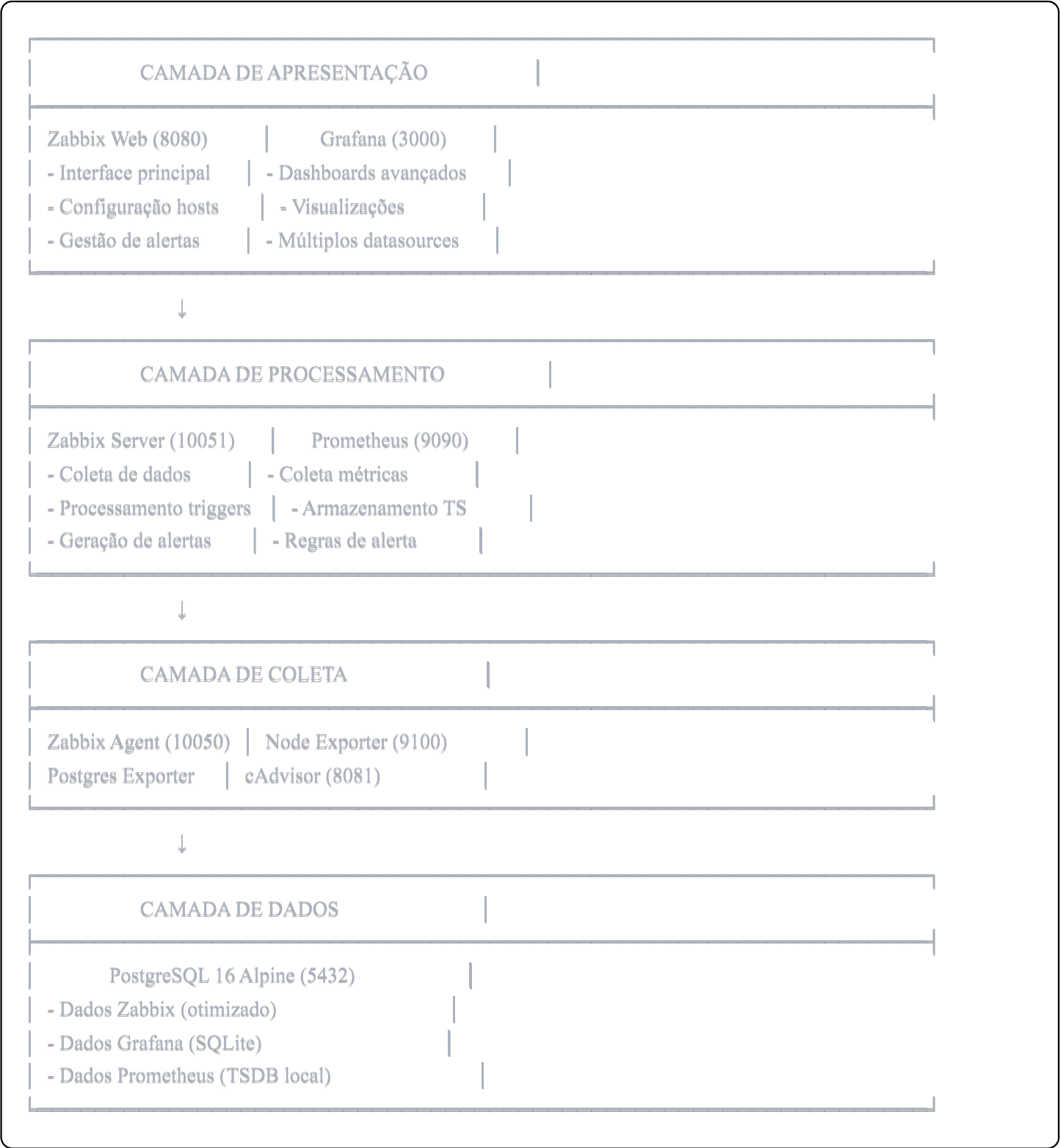
Componente	Versão	Porta	Função
Zabbix Server	7.4	10051	Core de monitoramento
Zabbix Web	7.4	8080	Interface web
Zabbix Agent 2	7.4	10050	Agent local
PostgreSQL	16-alpine	5432	Banco de dados
Grafana	latest	3000	Dashboards e visualização
Prometheus	latest	9090	Coleta de métricas
Node Exporter	latest	9100	Métricas do sistema
Postgres Exporter	latest	9187	Métricas do PostgreSQL
cAdvisor	latest	8081	Métricas dos containers

Recursos:

-  Monitoramento de infraestrutura completo
-  Dashboards profissionais no Grafana

- ☒ Alertas configurados no Prometheus
- ☒ Scripts de manutenção automatizados
- ☒ Backup automatizado
- ☒ Health checks
- ☒ Otimizado para alto volume

Arquitetura



Estrutura de Arquivos

```
zabbix-stack/
├── .env                # Variáveis de ambiente
├── docker-compose.yml  # Definição dos serviços
├── grafana/           # Configurações Grafana
│   ├── dashboards/    # Dashboards JSON
│   │   ├── postgres-monitoring.json  # Dashboard PostgreSQL
│   │   ├── zabbix.json               # Dashboard Zabbix Stack
│   │   ├── docker-containers.json    # Dashboard Containers
│   │   └── nodeexporter.json          # Dashboard Sistema
│   ├── provisioning/  # Provisioning automático (opcional)
│   │   ├── dashboards/ # Configuração de dashboards
│   │   └── datasources/ # Configuração de datasources
│   └── scripts/        # Scripts de manutenção
│       ├── backup.sh   # Backup automatizado
│       ├── cleanup.sh  # Limpeza e manutenção (opcional)
│       └── healthcheck.sh # Verificação de saúde
├── postgres/          # Configurações PostgreSQL
│   └── backups/        # Backups do banco (gerados automaticamente)
├── prometheus/        # Configurações Prometheus
│   ├── prometheus.yml  # Configuração principal
│   ├── alerts/         # Regras de alerta
│   └── alerts.yml       # Definição de alertas
└── scripts/           # Scripts de manutenção
    ├── backup.sh       # Backup automatizado
    ├── cleanup.sh      # Limpeza e manutenção (opcional)
    └── healthcheck.sh   # Verificação de saúde
```

Nota: A pasta `zabbix/` foi removida pois continha apenas subpastas vazias não utilizadas pelo stack.

Serviços

PostgreSQL

yaml

Imagem: postgres:16-alpine

Hostname: postgres-zabbix

Porta: 5432

Recursos: 4 CPUs, 6GB RAM

Função: Banco de dados principal

Otimizações:

- `max_connections`: 300
- `shared_buffers`: 1GB
- `effective_cache_size`: 3GB
- `maintenance_work_mem`: 512MB
- `work_mem`: 10MB

Zabbix Server

yaml

Imagem: zabbix/zabbix-server-pgsql:alpine-7.4-latest

Hostname: zabbix-server

Porta: 10051

Recursos: 4 CPUs, 4GB RAM

Função: Core de monitoramento

Otimizações:

- `StartPollers`: 40
- `CacheSize`: 1G
- `HistoryCacheSize`: 512M
- `ValueCacheSize`: 512M
- `TrendCacheSize`: 256M

Zabbix Web

yaml

Imagem: zabbix/zabbix-web-nginx-pgsql:alpine-7.4-latest

Hostname: zabbix-web

Portas: 8080 (HTTP), 8443 (HTTPS)

Recursos: 2 CPUs, 1GB RAM

Função: Interface web

Zabbix Agent 2

yaml

Imagem: zabbix/zabbix-agent2:alpine-7.4-latest

Hostname: \${HOSTNAME} (do .env)

Porta: 10050

Função: Monitoramento local

Grafana

yaml

Imagem: grafana/grafana:latest

Hostname: grafana

Porta: 3000

Banco: SQLite (interno)

Função: Dashboards e visualizações

Plugins Instalados:

- alexanderzobnin-zabbix-app
- grafana-piechart-panel
- grafana-clock-panel

Prometheus

yaml

Imagem: prom/prometheus:latest

Hostname: prometheus

Porta: 9090

Retenção: 30 dias

Função: Coleta e armazenamento de métricas

Node Exporter

yaml

Imagem: prom/node-exporter:latest

Hostname: node-exporter

Porta: 9100

Função: Métricas do sistema operacional

Postgres Exporter

yaml

Imagem: prometheuscommunity/postgres-exporter:latest

Hostname: postgres-exporter

Porta: 9187

Função: Métricas do PostgreSQL

cAdvisor

yaml

Imagem: gcr.io/cadvisor/cadvisor:latest

Hostname: cadvisor

Porta: 8081

Função: Métricas dos containers Docker

Configuração

Arquivo .env

bash

Dominio

DOMAIN=suaempresa.com.br

Hostname do Zabbix Agent

HOSTNAME=zabbix-server-host

PostgreSQL

DB_USER=zabbix

DB_PASSWORD=SenhaSuperSegura123

Grafana

GRAFANA_ADMIN_USER=admin

GRAFANA_ADMIN_PASSWORD=Admin@Grafana2025

Primeiros Passos

1. Clonar/Criar estrutura:

bash

mkdir -p ~/zabbix-stack

cd ~/zabbix-stack

2. Criar pastas:

bash

```
mkdir -p grafana/{dashboards,provisioning/{datasources,dashboards}}
mkdir -p postgres/backups
mkdir -p prometheus/alerts
mkdir -p scripts
mkdir -p zabbix/{alertscripts,externalscripts,modules,enc,ssh_keys,mibs,snmptraps,ssl/{ca,certs,keys},web/modules}
```

3. Criar arquivos de configuração:

```
bash

# .env, docker-compose.yml, prometheus.yml, alerts.yml
# (usar os artefatos fornecidos)
```

4. Dar permissões:

```
bash

chmod +x scripts/*.sh
chmod 755 postgres/backups
chmod 755 grafana/provisioning
```

5. Subir o stack:

```
bash

docker compose up -d
```

6. Verificar:

```
bash

docker compose ps
./scripts/healthcheck.sh
```

Dashboards

1. PostgreSQL Monitoring

Arquivo: grafana/dashboards/postgres-monitoring.json

Painéis:

- Status do PostgreSQL (UP/DOWN)
- Conexões ativas

- Cache hit ratio
- Tamanho dos bancos
- Transações por segundo
- I/O do banco
- Operações (INSERT/UPDATE/DELETE/SELECT)
- TOP 10 maiores tabelas

Datasource: Prometheus

2. Zabbix Stack Overview

Arquivo: `grafana/dashboards/zabbix.json`

Painéis:

- Status dos serviços (Zabbix Server, Web, Agent, PostgreSQL)
- CPU usage por container
- Memory usage por container
- Network traffic
- Disk I/O
- Resource summary table

Datasource: Prometheus

3. Docker Containers Monitoring

Arquivo: `grafana/dashboards/docker-containers.json`

Painéis:

- CPU usage por container (%)
- Memory usage por container
- Network traffic (RX/TX)
- Disk I/O (Read/Write)
- Resource summary
- Memory distribution (pie chart)
- CPU distribution (pie chart)

Datasource: Prometheus

4. Sistema Operacional - Node Exporter

Arquivo: `grafana/dashboards/nodeexporter.json`

Painéis:

- CPU usage total (gauge)
- Memory usage (gauge)
- Disk usage (gauge)
- Uptime
- CPU por core
- Memory details
- Network traffic por interface
- Disk I/O por dispositivo
- System load average (1m, 5m, 15m)
- Filesystem usage (tabela)

Datasource: Prometheus

Importar Dashboards

Via Web UI:

1. Grafana → + → **Import**
2. Upload JSON ou cole o conteúdo
3. Selecione datasource **Prometheus**
4. **Import**

Via Provisioning:

```
bash
```

```
# Dashboards são carregados automaticamente de:  
grafana/dashboards/*.json
```

Manutenção

Scripts Disponíveis

1. healthcheck.sh

Função: Verificação completa de saúde do stack

Uso:

```
bash  
  
./scripts/healthcheck.sh
```

O que verifica:

- Status de todos os containers
- Portas em escuta
- Espaço em disco
- Uso de memória
- Volumes Docker
- Erros recentes nos logs
- Status do PostgreSQL
- Tamanho dos bancos
- Conexões ativas

Executar: Diariamente ou após mudanças

2. cleanup.sh

Função: Limpeza e manutenção do banco

Uso:

```
bash  
  
./scripts/cleanup.sh
```

O que faz:

- VACUUM no PostgreSQL
- REINDEX (opcional)
- Limpar logs antigos dos containers
- Limpar imagens Docker não utilizadas
- Remover volumes órfãos
- Estatísticas das tabelas

- Verificar dead tuples
- Backup opcional antes da manutenção
- Rotação de backups (mantém 7 dias)

Executar: Semanalmente

3. backup.sh

Função: Backup automatizado

Uso:

```
bash
./scripts/backup.sh
```

O que faz:

- Backup do PostgreSQL (compactado)
- Backup das configurações do Grafana
- Rotação automática (30 dias)
- Verificação de integridade
- Estatísticas de espaço

Executar: Diariamente (automatizar com cron)

Automatizar com Cron

```
bash

# Editar crontab
crontab -e

# Adicionar linhas:
# Health check diário às 8h
0 8 * * * cd /caminho/para/zabbix-stack && ./scripts/healthcheck.sh >> /var/log/zabbix-health.log 2>&1

# Backup diário às 2h
0 2 * * * cd /caminho/para/zabbix-stack && ./scripts/backup.sh >> /var/log/zabbix-backup.log 2>&1

# Cleanup semanal (domingo às 3h)
0 3 * * 0 cd /caminho/para/zabbix-stack && ./scripts/cleanup.sh >> /var/log/zabbix-cleanup.log 2>&1
```

Troubleshooting

Container não inicia

```
bash

# Ver logs
docker logs <container_name>

# Ver últimas 50 linhas
docker logs <container_name> --tail 50

# Seguir logs em tempo real
docker logs -f <container_name>

# Verificar configuração
docker inspect <container_name>
```

Zabbix Server não conecta ao banco

```
bash

# Testar conexão
docker exec postgres-zabbix psql -U zabbix -d zabbix -c "SELECT version();"

# Verificar senha no .env
cat .env | grep DB_PASSWORD

# Ver logs do Zabbix Server
docker logs zabbix-server | grep -i error
```

Grafana não mostra dados

```
bash

# Verificar datasources
# Grafana → Configuration → Data Sources

# Testar no Explore
# Grafana → Explore → Prometheus → Query: up
```