

Zabbix

INSTALAÇÃO EM CLUSTER DOCKER/SWARM/GLUSTERFS

CASE DE SUCESSO: TCE/RO

Atila Aloise de Almeida

Mail: atila.aloise@tce.ro.gov.br

github.com/atilaloise



Quem sou eu?

- Pai do Eduardo e da Luisa;
- Assistente de TI – Tribunal de Contas do Estado de Rondônia;
- Consultor de Infraestrutura de TI – Foco em software livre;
- Entusiasta/Evangelizador Open Source/Devops;
- Músico;
- Eterno aprendiz ;



Experiências com Zabbix

- Primeiro contato em 2012;
- Case de sucesso: Tribunal de Contas do Estado de Rondônia
- Instalação como Micro Serviço em cluster Docker/Swarm + GlusterFS
- Monitoramento de infraestrutura “Convencional”:
 - Bancos de Dados;
 - Windows, Linux;
 - Dispositivos de rede (links, switch, etc);
 - Monitoramento de Micro serviços;
 - Docker/Swarm

Agenda da Apresentação

- O Tribunal de Contas do Estado de Rondônia;
- Aderindo a cultura DevOps;
- A base para instalação do Zabbix;
- Componentes do Zabbix como micro serviços;
- Publicando o Zabbix com um Frontend para Micro serviços;
- Instalação e configuração dos agentes;
- Adaptações necessárias;
- To do List;

SETIC

Tribunal de Contas do Estado de Rondônia

MISSÃO:

PROVER TECNOLOGIA DA INFORMAÇÃO E COMUNICAÇÃO PARA O TCE-RO PROPORCIONANDO MECANISMOS PARA FISCALIZAÇÃO DA APLICAÇÃO DOS RECURSOS PÚBLICOS.

Aderindo a cultura Devops

- Por onde começar?
- O que eu sei, somado ao que você sabe, pode ser fenomenal;
- Trabalho repetitivo? Considere isso como um problema. Resolva!
- Seu tempo deve ser usado para desenvolver habilidades;
- Git Hub: Isso sim é uma rede social. Siga os projetos que lhe interessam;
- Use o git pra tudo!
- Leia os fontes, scripts e instruções;
- Perceba quanta coisa está ao seu alcance;
- Contribua, ensine, divulgue, desenvolva, documente ...

A base para instalação do Zabbix

Sistema Operacional:

- Ubuntu 16 - Net install

Cluster de disco:

- Volume GlusterFs replicado em todos os nós do swarm

Cluster docker:

- Seis Swarm Nodes;

Exemplo Fstab;

```
SRV-INFRA-DOCKER01:/dev /mnt/dev  
glusterfs  
defaults,_netdev,backupvolfile-  
server=SRV-INFRA-  
DOCKER02,backupvolfile-server=SRV-  
INFRA-DOCKER03,backupvolfile-  
server=SRV-INFRA-  
DOCKER04,backupvolfile-server=SRV-  
INFRA-DOCKER05,backupvolfile-  
server=SRV-INFRA-DOCKER06 0 0
```


Componentes do Zabbix como micro serviços

- Banco de dados Mysql:

mysql:5.7

- Parâmetros:

Charset = UTF8

Collation = UTF8_bin

- Arquivo de Configuração:

max_allowed_packet=1024M

thread_cache_size=256

max_connections=512

query_cache_limit=512M

query_cache_size=4048M

query_cache_type=1

innodb_buffer_pool_size=5096M

Componentes do Zabbix como micro serviços

- Criando Swarm Service de banco de dados para o Zabbix

```
docker service create --name ZabbixDB-mysql -t --network zabbix \
    -e MYSQL_DATABASE="zabbix" \
    -e MYSQL_USER="zabbix" \
    -e MYSQL_PASSWORD="zabbix" \
    -e MYSQL_ROOT_PASSWORD="root" \
    --mount type=bind,source=/mnt/infra/databases/conf.d/zabbix-
mysql.conf,destination=/etc/my.cnf \
    --mount type=bind,source=/mnt/infra/databases/zabbix,destination=/var/lib/mysql \
    -d mysql:5.7 --character-set-server=utf8 --collation-server=utf8_bin
```

Componentes do Zabbix como micro serviços

- Criando Swarm Service de SNMP TRAPPER para o Zabbix

```
docker service create --name ZabbixTrapper -t --network zabbix \
--mount type=bind,source=/mnt/infra/logs/zabbixtraps,destination=/var/lib/zabbix/snmptraps \
--mount type=bind,source=/mnt/infra/MIBS,destination=/var/lib/zabbix/mibs \
-p 162:162/udp -d zabbix/zabbix-snmptraps:ubuntu-trunk
```

Componentes do Zabbix como micro serviços

- **Criando Swarm Service de JAVA GATEWAY para o Zabbix**

```
docker service create --name Zabbix-java-GW -t --network zabbix \  
    -e TZ="America/Porto_Velho" \  
    -p 10052:10052 \  
    -d zabbix/zabbix-java-gateway:latest
```

Componentes do Zabbix como micro serviços

Zabbix Server:

Integrações:

- **Traps Snmp**
- **Mibs**
- **External scripts:**
- **Suporte a integração com Vmware**

Componentes do Zabbix como micro serviços

- Criando Micro serviço no swarm:

```
docker service create --name ZabbixSRV -t --network zabbix --constraint 'node.id == ryo63cacq6jz9edban96ntjta' \
-e DB_SERVER_HOST="ZabbixDB-mysql" -e MYSQL_DATABASE="zabbix" \
-e MYSQL_USER="zabbix" -e MYSQL_PASSWORD="zabbix" \
-e MYSQL_ROOT_PASSWORD="root" -e ZBX_JAVAGATEWAY="Zabbix-java-GW" \
-e ZBX_TIMEOUT="30" -e ZBX_STARTVMWARECOLLECTORS=1 \
-e ZBX_VMWAREFREQUENCY=60 -e ZBX_VMWAREPERFFREQUENCY=60 \
-e ZBX_VMWARECACHESIZE=128M -e ZBX_VMWARETIMEOUT=10 \
-e StartPollers="80" -e StartPingers="10" \
-e SetartPollersUnreachable="80" -e StartIPMIPollers="10" \
-e StartTrappers="20" -e StartDBSyncers="8" \
-e ZBX_CACHESIZE="32M" -e ZBX_TRENDCACHESIZE="64M" \
-e ZBX_ENABLE_SNMP_TRAPS="true" \
--mount type=bind,source=/mnt/infra/logs/zabbixtraps,destination=/var/lib/zabbix/snmptraps \
--mount type=bind,source=/mnt/infra/MIBS,destination=/var/lib/zabbix/mibs \
--mount type=bind,source=/mnt/infra/zabbix/externalscripts,destination=/usr/lib/zabbix/externalscripts \
-p 10051:10051 -d zabbix/zabbix-server-mysql:latest
```

Componentes do Zabbix como micro serviços

Zabbix Web App (NGINX):

Integrações:

- **Traps Snmp**
- **Mibs**
- **TRAEFIK FRONTEND**

Componentes do Zabbix como micro serviços

Zabbix Web App (NGINX):

```
docker service create --name ZabbixWEB -t --network zabbix \
  -e PHP_TZ="America/Porto_Velho" \
  -e DB_SERVER_HOST="ZabbixDB-mysql" \
  -e MYSQL_DATABASE="zabbix" \
  -e MYSQL_USER="zabbix" \
  -e MYSQL_PASSWORD="zabbix" \
  -e MYSQL_ROOT_PASSWORD="root" \
  -e ZBX_SERVER_HOST="ZabbixSRV" \
  -e ZBX_SERVER_NAME="MONITORAMENTO TCE-RO" \
  -e ZBX_MEMORYLIMIT="1024M" \
  -e ZBX_MAXEXECUTIONTIME="1000" \
  -e ZBX_MAXINPUTTIME="1000" \
  -l "traefik.port=80" -l "traefik.frontend.rule=Host:zabbix.tce.ro.gov.br" \
  -l "traefik.docker.network=zabbix" -l "traefik.frontend.redirect.entryPoint=https" \
  -d zabbix/zabbix-web-nginx-mysql:latest
```


- Nada que um apt-get não resolva;
- Exemplo de template Puppet;

EnableRemoteCommands=1

LogRemoteCommands=1

AllowRoot=1

Server=<%= zabbix_server %>

Hostname=<%= hostname.split('.').first%>

Include=<%= config_dir %>/zabbix_agentd.d/

ServerActive=<%= zabbix_server %>

Instalação e configuração dos agentes

- O contêiner ZabbixSRV deve ser visto pelos agentes com o IP do Docker Host que o estiver rodando;

```
iptables -t nat -I POSTROUTING -s $(ip_CONTAINER) -j SNAT --to-source $(ip_HOSTDOCKER)
```

- No agente zabbix do Host docker que rodar o contêiner do ZabbixSrv, a conf "Server=" deve apontar pro ip do contêiner;

Adaptações necessárias

O Zabbix Server em contêiner se comporta de forma diferente em relação a rede.

- Permitir flutuação do Zabbix Server pelos nós do cluster sem precisar alterar configurações dos agentes;
- Perguntas?
- Sugestões?

To do List:

OBRIGADO!