ESCOLA SENAI INFORMÁTICA

Davi Abade

Gustavo Lima

João Pedro Araújo

Nicoly Zillig Dias

Maxuell Chaves

Roteiro para monitoramento

SÃO PAULO

2022

Davi Abade

Gustavo Lima

João Pedro Araújo

Nicoly Zillig Dias

Maxuell Chaves

Roteiro para monitoramento

Documento de passo a passo para configurar monitoramento da AWS via Zabbix.

SÃO PAULO

2022

SUMARIO

# INTRODUÇÃO

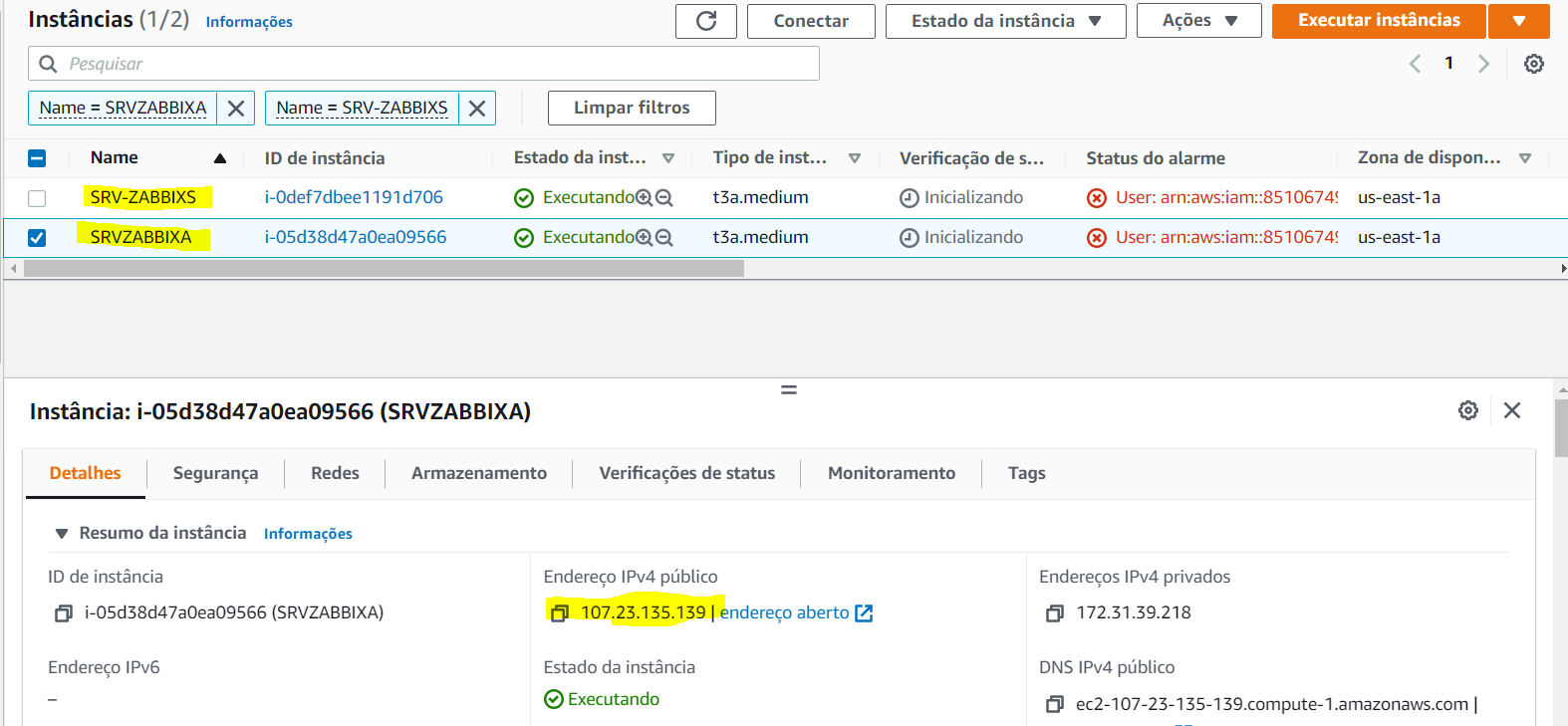
Este documento tem como objetivo explicar como é feito o monitoramento da AWS via Zabbix, demonstrando passo a passo as configurações necessárias e ferramentas que são utilizadas.

# REQUISITOS DE AMBIENTE

* Em primeiro lugar, é necessário instalar o [AWS CLI](https://awscli.amazonaws.com/AWSCLIV2.msi).
* Possuir acessoa permissões do [IAM](https://aws.amazon.com/pt/iam/features/manage-permissions/) e [AWS Auto Scaling](https://aws.amazon.com/pt/autoscaling/).
* Um servidor e agente [ZABBIX](https://www.zabbix.com/).

# CONFIGURAÇÕES

* Rodar instância na AWS

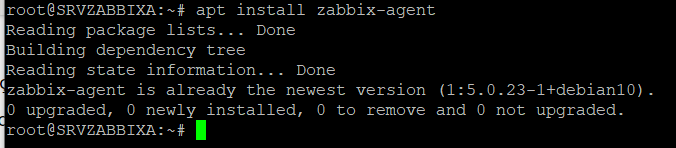


* Instalar os pacotes de serviço do Zabbix via CLI.

Wget https://repo.zabbix.com/zabbix/5.0/debian/pool/main/z/zabbix-release/zabbix-release\_5.0-1+buster\_all.deb

* Instalar Zabbix Agent

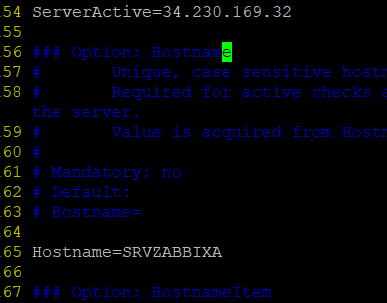
apt install zabbix-agent



* Configurar Hostname, Ip do Server e Server ativo do Zabbix no arquivo zabbix\_agentd.conf

vim /etc/zabbix/zabbix\_agentd.conf

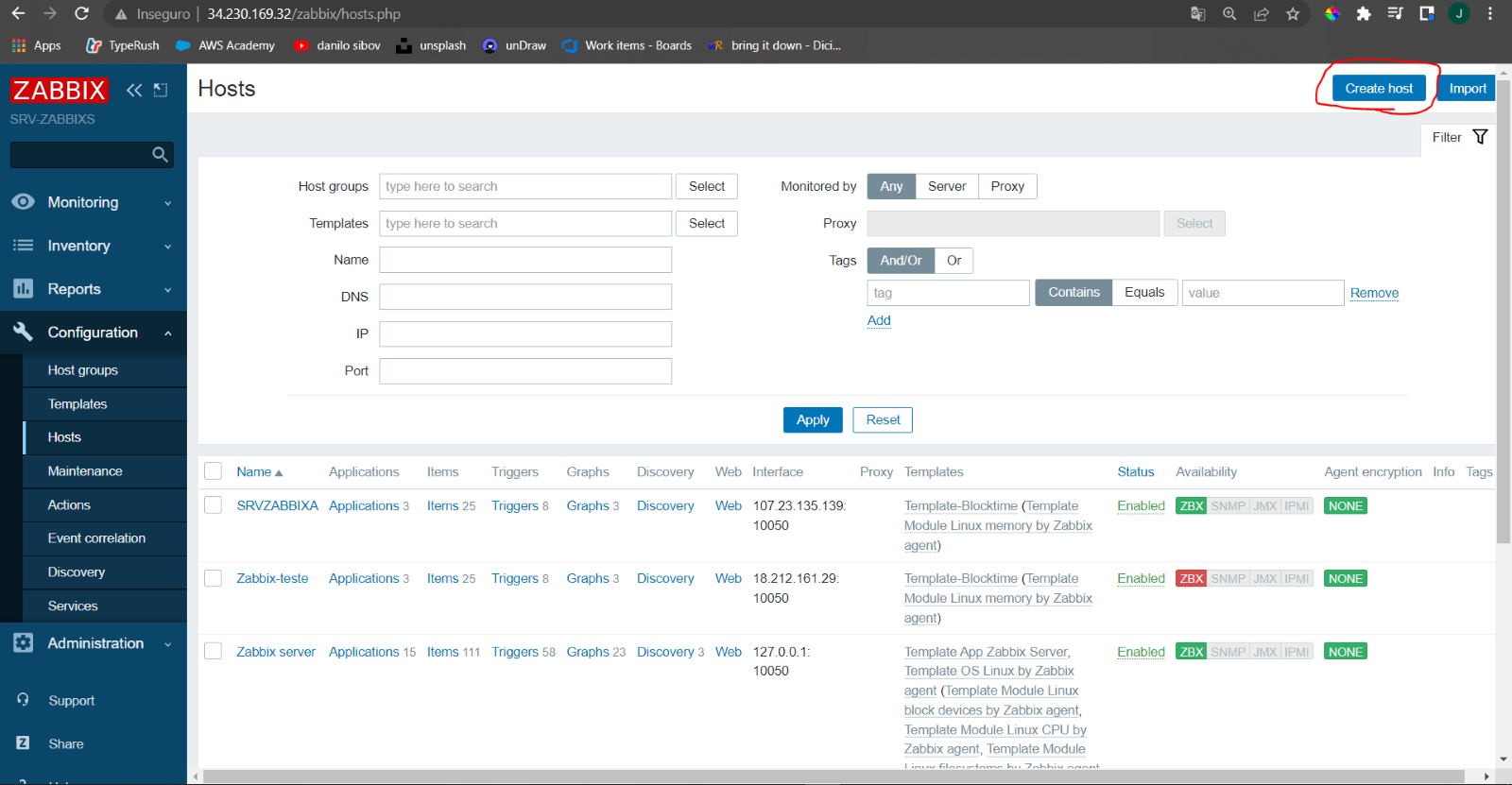
C:\Users\43017102807\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\2CF8D390.tmp

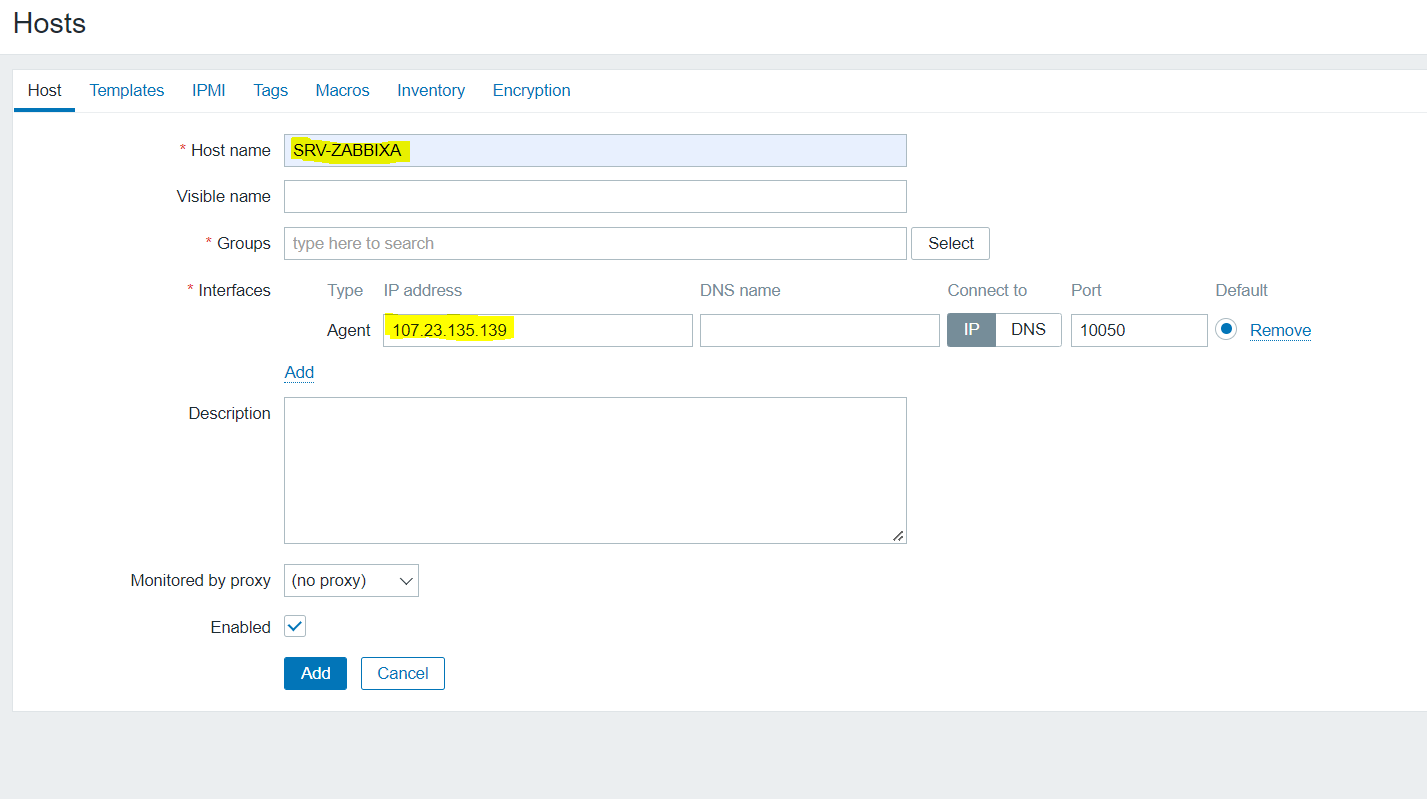


Obs: Após configuração do arquivo de configuração do zabbix-agent reinicie o serviço.

Systemctl restart zabbix-agent

* Criar uma nova Host no painel do Zabbix passando o Ip publico da AWS





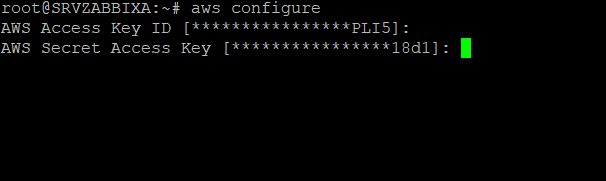
* Reiniciar o serviço

service zabbix-agent restart

* Instalar os pacotes do [AWS CLI](https://docs.aws.amazon.com/pt_br/cli/latest/userguide/getting-started-install.html)

* Informar aws\_access\_key\_id e aws\_secret\_access\_key do IAM no AWS CLI

aws configure



* Liberar os scripts remotos no arquivo zabbix\_agentd.conf

vim /etc/zabbix/zabbix\_agentd.conf

EnableRemoteCommands=1

LogRemoteCommands=1

DenyKey=system.run[]

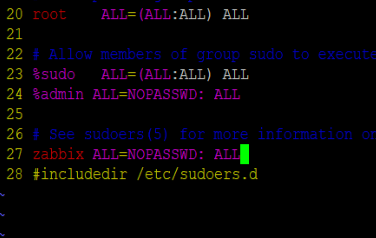
AllowKey=system.run[ \*]

* Liberar o sudo no Zabbix

vim /etc/sudoers

Escreva *“zabbix ALL=NOPASSWD: ALL”* na linha 27 do repositório

## Configurações dos scripts



* Criar uma pasta na qual serão armazenados os scripts

*mkdir /etc/zabbix/scripts*

C:\Users\43017102807\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\54E476B4.tmp

* Criar Scripts que serão atrelados aos itens do Checklist

Exemplo:

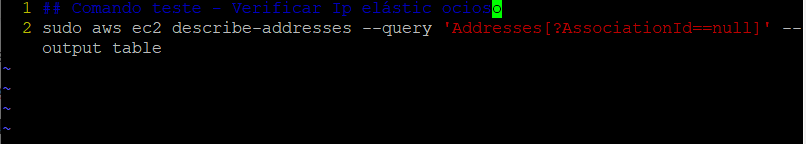
* Criar um arquivo do script dentro da pasta ‘Scripts’

*vim /etc/zabbix/scripts/exemploscript*

C:\Users\43017102807\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\48F472E2.tmp

* Colocar o comando dentro do arquivo

Obs: Os comandos da AWS precisam ter o *“sudo”* antes.



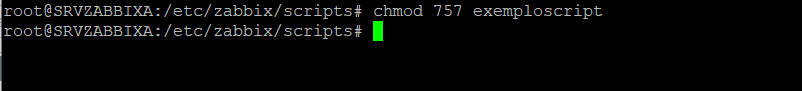
sudo aws ec2 describe-addresses --query 'Addresses[?AssociationId==null]' --output table

* Sair e salvar

*:Wq*

Obs: a pasta na qual está localizada os scripts devem ter permissão 757 ou 777

chmod 757 exemploscript



Repita isso com todos os scripts

# SCRIPTS DO CHECKLIST

## Verificações genéricas na console aws

### Verificar se possui S3. ✅

sudo aws s3 ls | wc -l

### Verificar se possui versionamento no S3.

(manual)

### Verificar destinatários SNS

\*

### Verificar usuários e políticas IAM ✅

aws iam list-users

### Verificar valores das faturas (Comparar últimas)

(manual)

### Possui faturas pendentes?

(manual)

## Identificação do servidor

### Verificar a execução dos snapshots diários. ✅

data=(date +'%Y-%m-%d')

aws ec2 describe-snapshots --query "Snapshots[?(StartTime>='{data}')].{Time:StartTime}" | grep -v "[" | grep -v "]" | grep -v "{" | grep -v "}," | grep -v "}" | wc -l

### Verificar se possui volumes sem uso. ✅

aws ec2 describe-volumes --filters Name=status,Values=available

### Verificar instâncias reservadas. ✅

aws ec2 describe-reserved-instances

### Verificar Elastic IP ocioso. ✅

aws ec2 describe-addresses --query 'Addresses[?AssociationId==null]' --output table

### Verificar Security Group (EC2 + RDS ) ✅

aws ec2 describe-security-groups --output table

aws rds describe-db-security-groups --output table

### Verificar opção Autoscalling (Habilitada ou não) ✅

aws autoscaling describe-auto-scaling-instances

### CloudWatch, alertas configurados, se está “OK”?

(manual)

### CloudWatch, alertas configurados, se está “Insufficient”? Corrigir.

(manual)

### CloudWatch, alertas configurados, se está Erro? Corrigir.

(manual)

## Verificações SSH

### Verificar se possui upgrades disponíveis (apt list --upgradable) - Corrigir com script ✅

apt list --upgradable

### Verificar reinicialização pendente - Corrigir com script ✅

if ps -C shutdown > /dev/null; then

echo "Shutdown is pending"

else

echo "Shutdown is not scheduled"

fi

### Verificar espaço em disco (df -h). ✅

df -h

### Verificar uso de memória ram/swap (free -h) ✅

free -h

### Validar antivírus

(manual)

### Validar fuso horário (date) ✅

date

### Verificar Crons ✅

cat /etc/crontab

### Verificar backups dos dias da semana ✅

cat /etc/crontab

### Analisar logs de acesso Apache (buscar URL/HTTP/Code) ✅

tail -f /var/log/apache2/access.log

### Analisar logs de erro Apache, sinalizar e enviar ao cliente ✅

tail -f /var/log/apache2/error.log

# userparameter

**UserParameter** é uma forma de estender as funcionalidades do Zabbix Agent. Com esta funcionalidade podemos criar novas chaves de itens de acordo com as nossas necessidades.

* Criar o arquivo userparameter

touch /etc/zabbix/zabbix\_agentd.d/userparemeter\_aws.conf

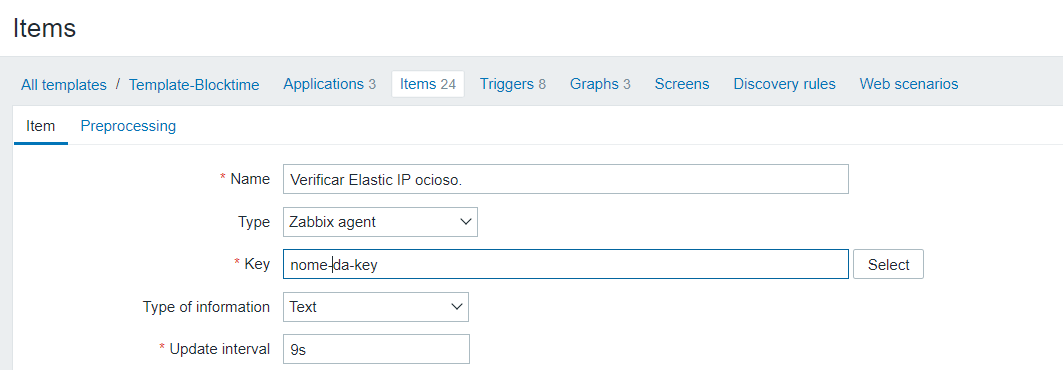
Após a criação dos scripts e da pasta ‘userparameter’, entre no arquivo **‘**userparameter’ e configure de acordo com o exemplo abaixo.

vim /etc/zabbix/zabbix\_agentd.d/userparemeter\_aws.conf

Exemplo:

Userparameter={nome-da-key},{/Caminho/exemploscript}

C:\Users\43017102807\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\A83BF68C.tmp



\*/etc/zabbix/scritps/exemploscript\*

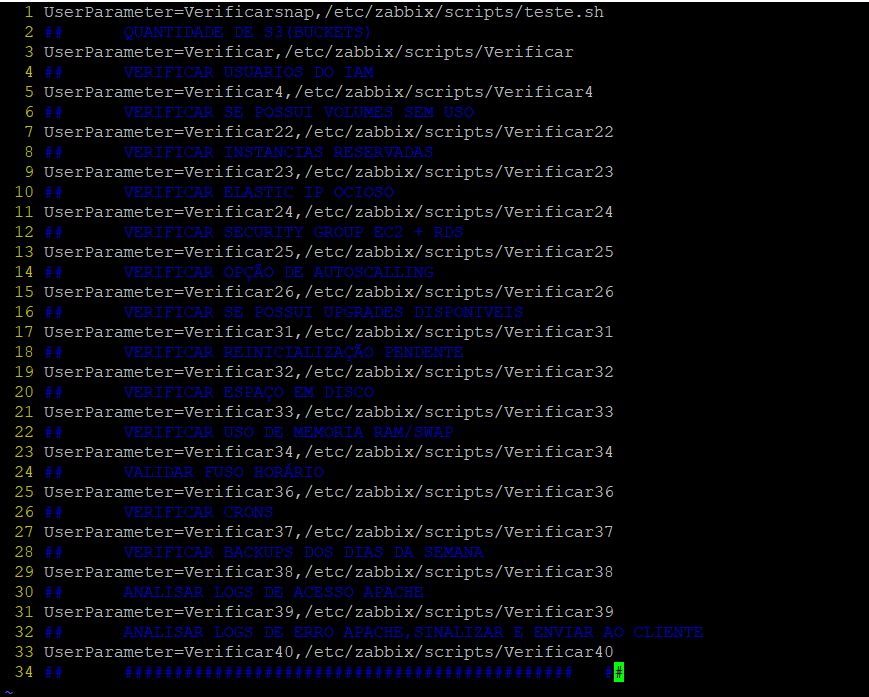
* Exemplo na prática

userparameter=exemploscript,/etc/zabbix/scripts/exemploscript

* Repita isso com todos os itens, se for o caso

## Arquivo userparameter do checklist

### No repositório vim:



ATENÇÃO: Salve e saia do arquivo (wq) e de o comando:

systemctl restart zabbix-agent.service

Para reiniciar o serviço zabbix-agent.

# Zabbix Trapper

Alguns scripts ultrapassam o limite máximo de tempo de resposta e com isso geram um timeout. Então, para resolver este problema precisamos utilizar o Zabbix Trapper.

Como pré-resquisito temos que instalar o Zabbix Sender no servidor, e para isso utilizamos o comando:

apt install zabbix-sender

C:\Users\43017102807\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\411B7164.tmp

* Para configurar o Zabbix Trapper temos que utilizar o comando:

crontab –e

Para definirmos o tempo que um comando será executado:

C:\Users\43017102807\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\A1D73112.tmp

Dentro do crontab temos que digitar o seguinte comando no final do arquivo:

\*/tempo(s) \* \* \* \* zabbix\_sender –z ip-do-zabbix-server -s “Nome-do-host” -k nome-da-chave  –o $(caminho-do-arquivo)

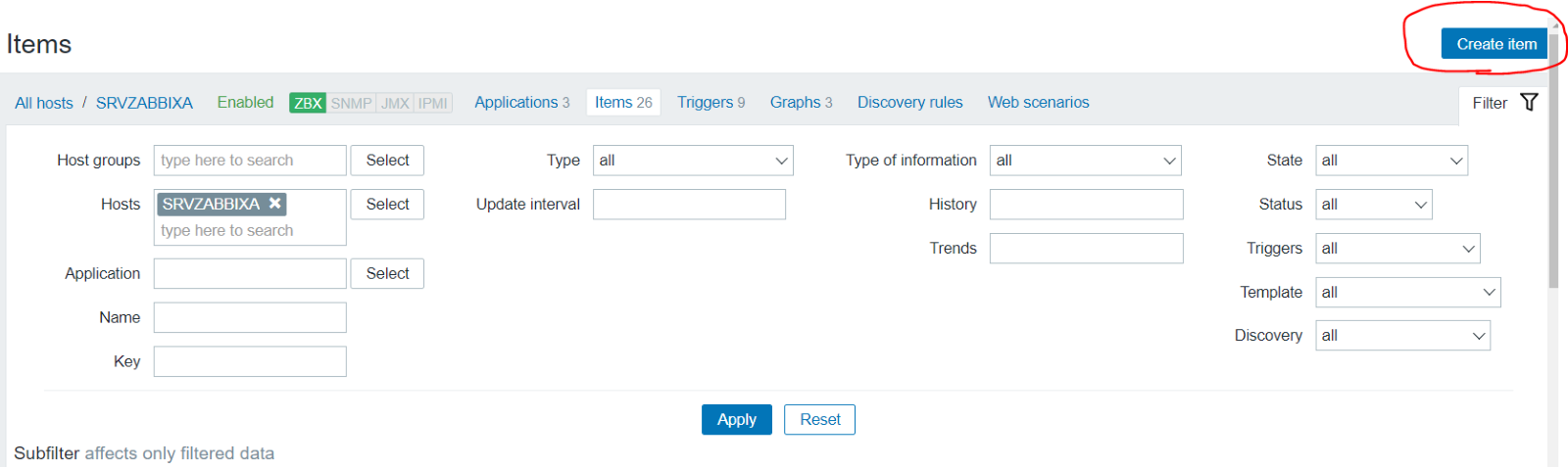
Exemplo:

\*/120 \* \* \* \* zabbix\_sender –z 34.230.169.32 -s “SRVZABBIXA” -k verificarSnap –o $(/etc/zabbix/scripts/Verificar21)

C:\Users\43017102807\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\86BB42F0.tmp

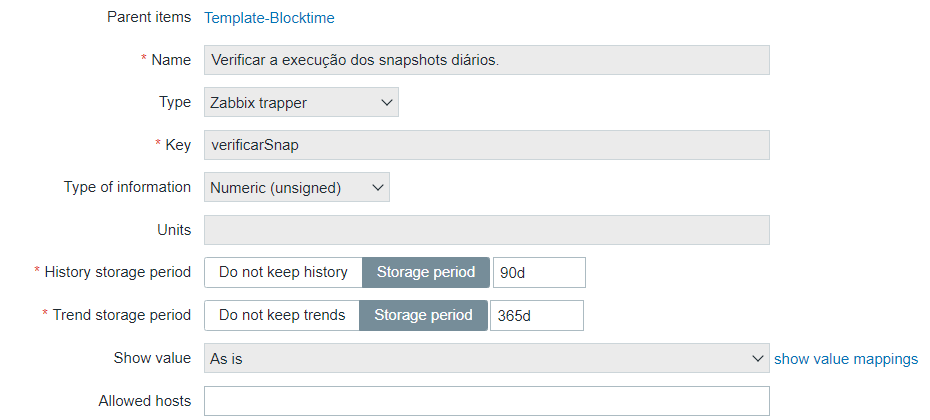
Em seguida, precisamos criar o **item** no Zabbix

**Clicamos** em **create** item



* **Configurações do item:**
* Colocamos o Nome Do Item, o Type (Zabbix Trapper) e a Key.

Obs: nome-da-key precisa estar configurada no Crontab.



Verificar se os dados estão sendo exibido em Latest Data (que fica em monitoramento) .

Um ponto importante é que os dados serão exibidos conforme o tempo que foi especificado no Crontab**,** caso vocês queiram acelerar esse tempo para que os dados sejam exibidos mais rápido temos que executar o comando no próprio terminal do servidor.

Por exemplo:

zabbix\_sender –z 34.230.169.32 -s “SRVZABBIXA” -k verificarSnap –o $(/etc/zabbix/scripts/Verificar21)

C:\Users\43017102807\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\3BD3FCAA.tmp

E depois é só verificar em Latest Data com o nome do item que foi criado, e os dados serão exibidos “rapidamente”

