

Olá pessoal, hoje vou falar de um micro serviço que criei para simplificar o uso do Zabbix sender, criando um webhook (API), para que você consiga enviar as suas métricas. Aqui vou colocar um “how to” de como seguir mas lembre, quanto mais experiência você tiver com o Zabbix maiores são as possibilidades desse método (Preprocessing!).

Mas o que é o Sender API for Zabbix

É um componente criado para facilitar o uso do recurso “sender” da Ferramenta de monitoramento Zabbix.

“O sender se trata de uma lib. que pode ser integrada diretamente no código para enviar métricas para o Zabbix”

Por muitas vezes é necessário fazer diversas homologações e testes para adicionar novos recursos para a aplicação o que pode inviabilizar uma integração com recursos terceiros ou em alguns casos quando skill técnicos são necessários.

Com o Sender API for Zabbix é necessário apenas apontar para o endpoint (API), disponibilizado passando algumas informações para o host/item

E o valor da métrica será enviado ao Zabbix sem a necessidade de lib. ou pacotes no código da sua aplicação, poderá ser chamado de qualquer lugar com suporte a http.

## **How to use**

É necessário ter um host e item do tipo zabbix trapper para ser passado no body ao realizar o post, a descrição está descrita abaixo:

Criar um host no Zabbix

- Em configurações / Hosts / Create host

**ZABBIX** << >> Zabbix - Filgs

**Hosts** Create host Import

Filter

Host groups: type here to search Select

Templates: type here to search Select

Name:

DNS:

IP:

Port:

Monitored by: Any Server Proxy

Proxy:  Select

Tags: And/Or Or

tag:  Contains Equals value:  Remove

Add

Apply Reset

Defina um nome e adicione a um grupo (demais opções conforme a sua necessidade).

- No exemplo estou colocando o nome Sender, ao finalizar clique em add.

## Hosts

**Host** Templates IPMI Tags Macros Inventory Encryption

\* Host name:

Visible name:

\* Groups: Virtual machines Select

\* Interfaces:

Type	IP address	DNS name	Connect to	Port	Default
Agent	127.0.0.1		<span>IP</span> <span>DNS</span>	10050	<input checked="" type="radio"/> <span>Remove</span>

Description:

Monitored by proxy: (no proxy)

Enabled: ☒

Add Cancel

Agora crie um template com o nome desejado.

- Em Templates/Create Template

**ZABBIX**
Zabbix - Filgs

Monitoring
Inventory
Reports
Configuration
Host groups
**Templates**
Hosts

## Templates

Template
Linked templates
Tags
Macros

\* Template name

Visible name

\* Groups

Select

Description

Add

Cancel

Acesse novamente o template e crie o item conforme imagens a seguir.

## Templates

Create item

Host groups

Select

Linked templates

Select

Name

Tags

And/Or Or

Contains Equals

Remove

Add

Apply

Reset

<input type="checkbox"/>	Name ▲	Hosts	Applications	Items	Triggers	Graphs	Screens	Discovery	Web	Linked templates	Linked to templates
<input type="checkbox"/>	sender_api	Hosts	Applications	Items	Triggers	Graphs	Screens	Discovery	Web		

Disp

## Create Item:

## Items

Create item

All templates / sender\_api
Applications
**Items**
Triggers
Graphs
Screens
Discovery rules
Web scenarios

Filter

Host groups

Select

Hosts

Select

Application

Select

Name

Key

Type

all

Type of information

all

State

all

Update interval

History

Trends

Status

all

Triggers

all

Template

all

Discovery

all

Apply

Reset

Subfilter affects only filtered data

<input type="checkbox"/>	Wizard	Name ▲	Triggers	Key	Interval	History	Trends	Type	Applications	Status	Info

No data found.

## Items

[All templates](#) / [sender\\_api](#) [Applications](#) [Items](#) [Triggers](#) [Graphs](#) [Screens](#) [Discovery rules](#) [Web scenarios](#)

[Item](#) [Preprocessing](#)

\* Name

meu item

Type

Zabbix trapper ▼

\* Key

sender\_metrica

Select

Type of information

Numeric (unsigned) ▼

Units

No exemplo acima incluímos o nome para o item, o tipo como Zabbix Trapper (que permite receber dados da API) e a chave (este que é o identificador para a métrica).

Tipo de informação pode ser definido conforme a sua necessidade o Sender API pode enviar Strings ou valores numéricos para o Zabbix, porem o mesmo será interpretado conforme o tipo definido, ressaltando que existem formas para alterar o tipo de dado e ou estrutura usando Preprocessing (não discutidos neste documento).

Agora adicione o template ao host criado no primeiro step.

## Hosts

[All hosts](#) / [Sender](#) [Enabled](#) [ZBX](#) [SNMP](#) [JMX](#) [IPMI](#) [Applications](#) [Items](#) [Triggers](#) [Graphs](#) [Discovery rules](#) [Web scenarios](#)

[Host](#) [Templates](#) [IPMI](#) [Tags](#) [Macros](#) [Inventory](#) [Encryption](#)

Linked templates

Name	Action
------	--------

Link new templates

sender\_api ✕

type here to search

Select

Update

Clone

Full clone

Delete

Cancel

Atualize e agora vamos usar o microserviço do Sender API, neste exemplo será configurado no Docker.

Faça o download da imagem no Docker Hub:

- docker push rafasera/sender-api-zbx:latest

 rafasera / sender-api-zbx

*This repository does not have a description* 

Inicie a mesma conforme imagem e descrição abaixo

```
rafael@Rafaels-MacBook-Pro ~ % docker run -d -p 80:80 -e API_URL="104.44.133.54" -e API_PORT=10051 rafasera/sender-api-zbx  
cedc758baa5b2db5ba4c789308dc3c1b5c8bd12c6529d4f1eddf89429c9bab4f  
rafael@Rafaels-MacBook-Pro ~ % █
```

Aqui foi exposta a porta 80

Em API\_URL informe o endereço do Zabbix.

Em API\_PORT informe a porta do seu Zabbix, default é 10051

Após isso aguarde inicializar.

“OBS você pode encontrar o mesmo tutorial acessando o container no caminho default que definiu no meu caso seria a localhost”

Por fim vamos utilizar o Sender-API

“Eu vou usar o Programa Postman para efetuar as requests”

Abra o Postman e acesse a rota:

- <https://filgs.com.br/api/v1/zabbix-sender>

Tipo:

- POST

Body:

```
{  
  "host": "Sender",  
  "item": "sender_metrica",  
  "value": "99"  
}
```

Obs.

- O endereço localhost foi atribuído pelo docker ressaltando que estou usando localmente.
- O body deve conter o host criado no Zabbix para receber a métrica
- A chave do Item (key)
- E o valor da métrica que pode ser enviado por outra ferramenta ou aplicação.

http://localhost/api/v1/zabbix-sender

POST http://localhost/api/v1/zabbix-sender

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "host": "Sender",
3   "item": "sender_metrica",
4   "value": "99"
5 }
```

Body Cookies Headers (7) Test Results

Status: 201 Created Time: 44 ms Size: 305 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "title": "Zabbix_Sender_API",
3   "version": "0.0.1",
4   "status": "sucess"
5 }
```

Latest data

Host groups type here to search Select Name

Hosts Sender X type here to search Select Show items without data

Application type here to search Select Show details

Apply Reset

Host	Name	Last check	Last value	Change
Sender	API (1 item)			
	meu item	2021-03-06 15:47:12	99	Graph

Displaying 1 of 1 found

Não havendo problemas de estrutura ou comunicação com o Sender-API receberá a mensagem de sucesso.

Em seguida a métrica enviada chegara no seu Zabbix.