



API calls and processing with jq



WEBINAR '24



EQUIPMENT



Emanuelle Ferreira



Fabrizio Luongo



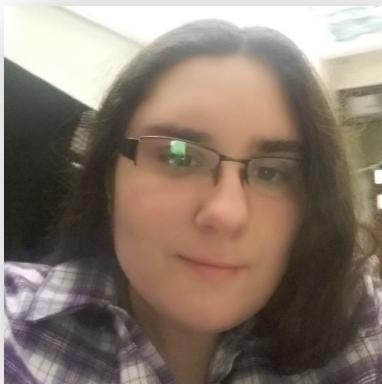
Gabriel Varela



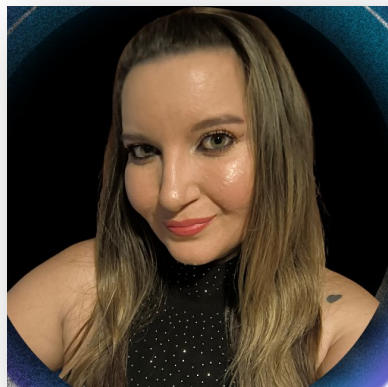
Gustavo Guido



Joaquin Gucci



Martina Romero



A. Soledad Turino

ZABBIX

ZABBIX
CERTIFIED PARTNER
ZABBIX
TRAINING PARTNER

WHAT WE OFFER

ZABBIX

Workshops and training



We promote your **autonomy** in the use of your monitoring solution.



Community
Zabbix Español

Consultancy

Follow the **change** in your organization's services and assets, incorporating them into your monitoring system, always having **everything under control**.



24x7 remote monitoring



Alarm monitoring service, **every day of the year**, performed remotely.



Support

Attention **without limit** of requests.

Turn Key

Gain speed to achieve monitoring of your services and assets.



Topics

1. Brief introduction of Zabbix API
2. Prepare the environment
 - a. Download the Zabbix Appliance or have your own installation
 - b. Clone the CZAPI repository
3. Generate the connection to the Appliance
4. Retrieve base information
 - a. Hosts
 - b. Templates
5. Modify the result
 - a. Remove columns from the list
 - b. Format the result to display it in a spreadsheet
 - c. Add information

API Application Program Interface

A set of functions to interact with an application

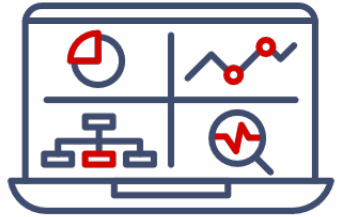
Zabbix offers a set of functions that allows you to do everything that can be done from the frontend.

To use the Zabbix API it is necessary to have access to the Zabbix Frontend URL and have credentials to enter, it can be a username and password or an API Token (recommended).

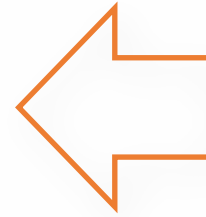
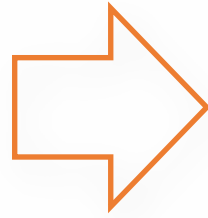
Zabbix API controls permissions in the same way that the Frontend does.

Brief introduction of Zabbix API

Zabbix API working.



JSON requirement



JSON response



Frontend

My computer

JSON Requirement - JSON specifying what we require

JSON Response - Request Result

Brief introduction of Zabbix API

API JSON Requirement

Request to Zabbix, information or modifications.

```
{  
  "jsonrpc" : "2.0",  
  "method" : "Name of the function",  
  "params" : { function parameters },  
  "auth" : "Authorization token",  
  "id" : 1  
}
```

Mandatory and fixed
An action to carry out
Can be an array
API token or login token
Mandatory but we do not use it

Brief introduction of Zabbix API

API JSON Requirement

"method" : " Object . Action "

Object : “Zabbix Concepts” **Action** : What we want to do

Host	get
hostgroup	delete
template	create
Item...	update...

Not all objects support all functions, for example, an event cannot be created, it will be a consequence of monitoring.

Brief introduction of Zabbix API



API JSON Requirement

"params" :{ function parameters }

get

If we are asking for information, the parameters will determine the filters.

update , delete ...

If we are modifying information, the parameters will be the **objects** that we are going to work with.

Brief introduction of Zabbix API



JSON Response **API**

```
{  
  "jsonrpc": "2.0",  
  "result": { action result },  
  "id": 1  
}
```

The answer comes in the result property, its content depends on the function
get requested **objects**
update , delete ... id of the **objects** we work with

Download the Zabbix Appliance or have an installation



- From the [Zabbix page](#), download .ovf and extract the file

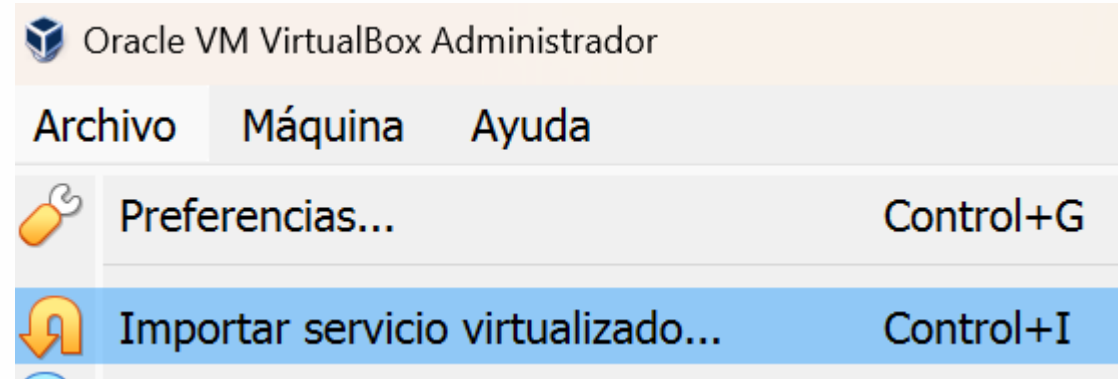
Zabbix Packages	Zabbix Cloud Images	Zabbix Containers	Zabbix Appliance	Zabbix Sources	Zabbix Agents
-----------------	---------------------	-------------------	-------------------------	----------------	---------------

Zabbix 7.0 LTS	7.0.0	Jun 04, 2024	Open virtualization format (.ovf)			SHA1 SHA256	Descargar
----------------	-------	--------------	-----------------------------------	--	--	----------------	-----------

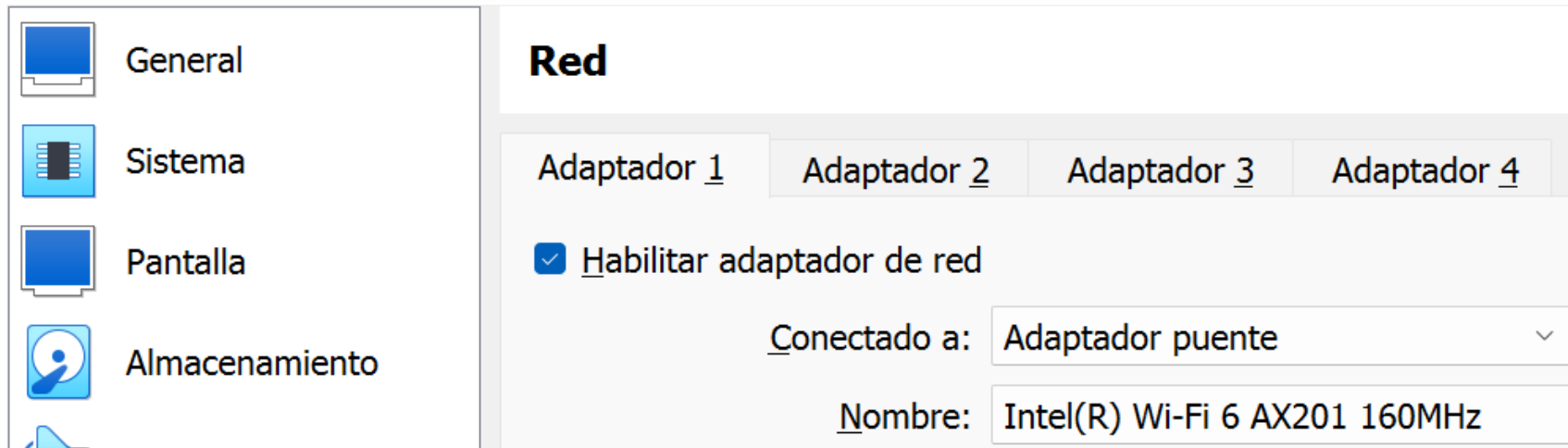
- Unzip the file

Download the Zabbix Appliance or have an installation

- Import virtualized service in VirtualBox



- Configure the VM Network to “Bridge Adapter”



Download the Zabbix Appliance or have an installation

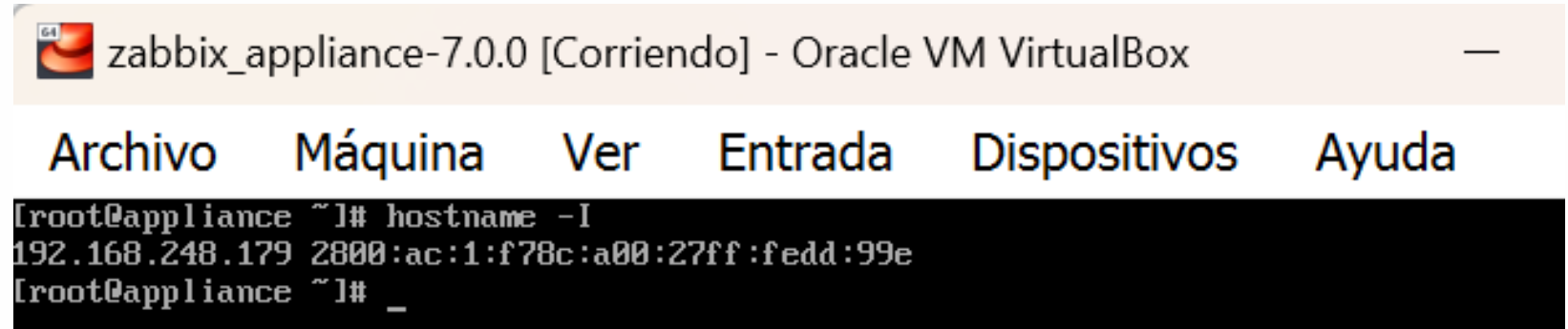
1. Login to the VM

a. root:zabbix

b. Find the IP

i. ip -a

hostname -I



```
zabbix_appliance-7.0.0 [Corriendo] - Oracle VM VirtualBox

Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

[root@appliance ~]# hostname -I
192.168.248.179 2800:ac:1:f78c:a00:27ff:fedd:99e
[root@appliance ~]# _
```

2. Log in from a terminal (WSL?)

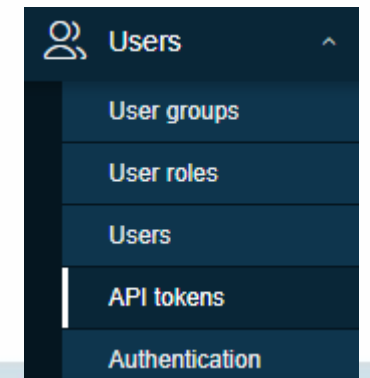
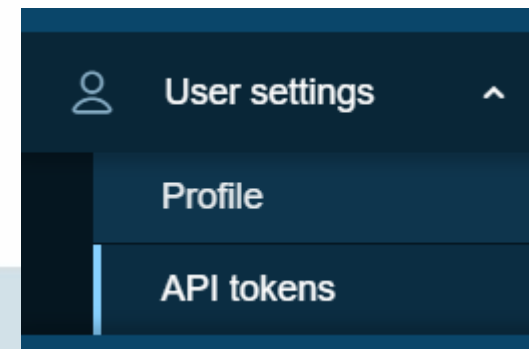
a. we gain facilities such as copy and paste, keyboard, etc.

b. ssh root@192.168.248.179


3. Enter the frontend of the virtual machine

a. Admin:zabbix

b. Generate an API token (save it)



Prepare the environment

1. Clone the CZAPI repository
 - a. It is recommended to update WLS
sudo apt update && sudo apt upgrade
 - b. Go to a directory where we are going to work
 - i. mkdir WebinarCustosAPI && cd WebinarCustosAPI
 - c. git clone <https://github.com/CUSTOSMonitoring/CZAPI.git>
 - d. A CZAPI directory was created
 - i. cd CZAPI
2. Prepare the environment
 - a. . SetEnvironment
 the point is to force this shell to run the cmd

Generate the connection to the Appliance



1. ZAPI -C W<tab>

- a. It will generate a file to connect to the Virtual Machine, a copy of an example, which it will complete with the <Tab>

2. Complete the options

- a. To do this, enter the name of the option (or part) or the number
- b. Options are already loaded
 - i. 5) DEF_FUNC
 - ii. 16) format
 - iii. 17) libreria (library)
- c. 12) UR L
 - i. [http:// 192.168.1.17](http://192.168.1.17) (the IP of the virtual machine, or our installation)
- d. 15) authToken **The token generated on the frontend**
- e. 1) Exit
 - i. Save the file, the enter will leave the same name, we overwrite the original

Hosts



1. ZAPI -c W<tab>
 - a. will complete with the example connection file
 - b. we can generate our own files to connect to other URLs and versions
2. ZAPI -c/WebinarApi7.0.zapi Host<tab><tab>
 - a. The possible functions will be shown
3. ZAPI -c/WebinarApi7.0.zapi HostG<tab>
 - a. Completed with HostGet
4. <Enter>
5. Shows the list of hosts defined in the Zabbix installation
 - a. The **codes are translated** so that they are understandable
 - i. this translation is configurable

Templates

1. ZAPI -c W<tab>
 - a. will complete with the connection file from the example
 - b. we can generate our own files to connect to other URLs and versions
2. ZAPI -c/WebinarApi7.0.zapi Templa<tab><tab>
 - a. The possible functions will be shown
3. ZAPI -c /WebinarApi7.0.zapi TemplateG<tab>
 - a. Completed with TemplateGet
4. <Enter>
5. Shows the list of templates defined in the Zabbix installation
 - a. The **codes are translated** so that they are understandable
 - b. this translation is configurable

Modify the result

Templates

1. To avoid specifying the configuration file every time
 - a. `export ZAPI_Conf=/PATH/AL/ARCH/CONF.zapi`
2. ZAPI TeplateGet
 - a. Observe the columns it recovers
 - i. many do not correspond to a template
 1. These types of errors, wishes etc. can be sent to [Zabbix](#)
 - ii. In specific listings we may not be interested in some

Modify the result

Remove columns from the list

1. ZAPI TemplateGet **output**=host,name,description,uuid,vendor_name,vendor_version

```
[ ...  
  {  
    "host": "Website by Browser",  
    "name": "Website by Browser",  
    "description": "The template to monitor a website's availability and performance...",  
    "uuid": "2526dce71d714e31bd545e96370c67b2",  
    "vendor_name": "Zabbix",  
    "vendor_version": "7.0-0",  
    "templateid": "10628"  
  } ...  
]
```

Modify the result

Format the result to display it in a spreadsheet

1. Let's generate a csv file
 - a. **jq** has “formatters”
 - i. `@csv` generates a “comma separated values”
 - ii. receives vectors with the values by columns
2. ZAPI TemplateGet **output**=host,name,description,uuid,vendor_name,vendor_version | \
jq ' **.[]** '
 - a. We “disassemble” the array, we have each template but not “in vector”
3. ZAPI TemplateGet **output**=host,name,description,uuid,vendor_name,vendor_version | \
jq ' **.[] | [.]** '
 - a. We have vectors but objects inside, we need strings
4. ZAPI TemplateGet **output**=host,name,description,uuid,vendor_name,vendor_version | \
jq ' **.[] | [.[]]** '
 - a. This one!

Modify the result

Format the result to display it in a spreadsheet

1. ZAPI TemplateGet output=host,name,description,uuid,vendor_name,vendor_version | \n jq ' .[] | [.[]] | @csv '
 - a. generates with many quote escapes
2. ZAPI TemplateGet output =host,name,description,uuid,vendor_name,vendor_version | \n jq -r ' .[] | [.[]] | @csv '
 - a. extra quotes removed
3. ZAPI TemplateGet output=host,name,description,uuid,vendor_name,vendor_version | \n jq -r ' .[] | [.[]] | @csv' > ReportTemplates.csv
4. Open the file in a “Spreadsheet” (libreoffice, google, excel)
 - a. Titles are missing
 - i. ZAPI... | jq -r ' (.[1] | keys), .[] | [.[]] | @csv' > ReportTemplates.csv
 - b. I give it a nice format

Modify the result

Add information

1. ZAPI HostGet **selectParentTemplates**

```
[....  
  {....  
    "monitored_by": "0",  
    "inventory_mode": "automatic",  
    "active_available": "1",  
    "assigned_proxyid": "0",  
    "parentTemplates" : [  
      ....  
    ]  
  }  
]
```


Modify the result

Add information

1. ZAPI HostGet **output=name,host** **selectParentTemplates=name,host**

```
[ {  
  "hostid": "10084",  
  "name": "Zabbix server",  
  "host": "Zabbix server",  
  "parentTemplates": [  
    {  
      "name": "Linux by Zabbix agent",  
      "host": "Linux by Zabbix agent"  
    }, ...  
  ]  
}]
```

Modify the result

Add information

```
1. ZAPI HostGet output=name,host selectParentTemplates=name,host | jq -r '[  
  "Host Visible Name","Host Name","Template Visible Name","Template Host" ],  
  (.[] |  
    [ .name, .host,  
      ( .parentTemplates[0] | .name ),  
      ( .parentTemplates[0] | .host )  
    ],  
    ( [ .parentTemplates [1:], .parentTemplates [1:] ] |  
      transpose | .[] | [ "", "", .[0].host, .[1].name ]  
    )  
  ] | @csv  
' > ReportHosts.csv
```

To learn more about jq
[jq manual](#)

To suggest new webinars, changes to existing ones...
capacitaciones@custos.uy

Enter the community, strength is born from collaboration, it can be [Telegram](#) in spanish,
[Zabbix users group](#) in English, among others

Questions



[/custos.uy](https://custos.uy)



info@custos.uy



[/custos-monitoring](https://www.linkedin.com/company/custos-monitoring)



WEBINAR

Thank you so much!

ZABBIX 

WEBINAR '24

ZABBIX

