



Llamadas a la API y procesamiento con jq



WEBINAR '24



EQUIPO



Emanuelle Ferreira



Fabrizio Luongo



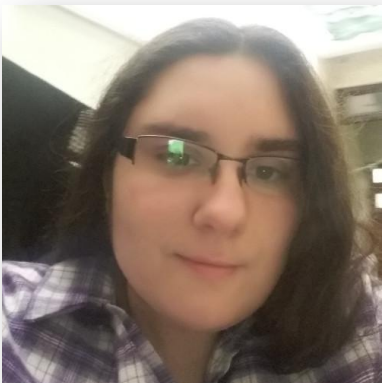
Gabriel Varela



Gustavo Guido



Joaquín Gucci



Martina Romero



A. Soledad Turino



QUE OFRECEMOS

ZABBIX

Talleres y capacitaciones



Favorecemos tu **autonomía** en el uso de tu solución de monitoreo.



Comunidad
Zabbix Español

Consultoría

Acompaña el cambio de los servicios y activos de tu organización, incorporándolos a tu sistema de monitoreo, teniendo siempre **todo bajo control**.



Monitoreo remoto 24x7



Servicio de monitoreo de alarmas, **todos los días del año**, realizado de manera remota.



Soporte

Atención **sin límite** de solicitudes.



Llave en mano

Gana velocidad para alcanzar la monitorización de tus servicios y activos.

Agenda

1. Breve introducción API de Zabbix
2. Preparar el ambiente
 - a. Bajar la Appliance de Zabbix o disponer de una instalación
 - b. Clonar el repositorio de CZAPI
3. Generar la conexión a la Appliance
4. Recuperar información base
 - a. Hosts
 - b. Templates
5. Modificar el resultado
 - a. Sacar columnas del listado
 - b. Formatear el resultado para levantarlo en una planilla de cálculo
 - c. Agregar información

API Application Program Interface

Un conjunto de funciones para interactuar con una aplicación

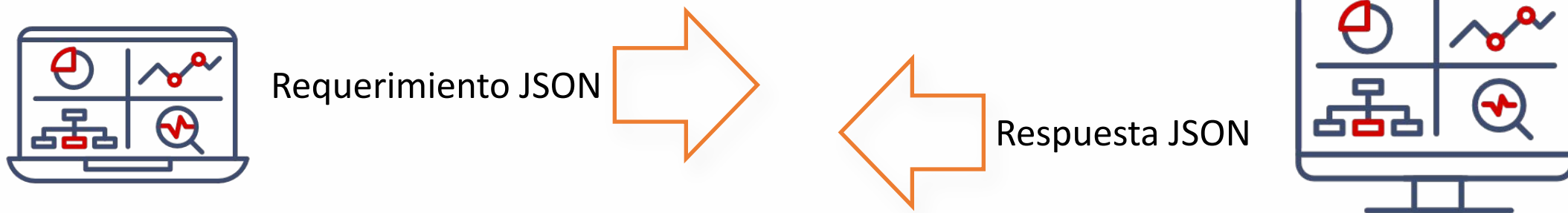
Zabbix ofrece un conjunto de funciones que permite realizar todo lo que se puede realizar desde el frontend.

Para utilizar la API de Zabbix es necesario tener acceso a la URL del Frontend de Zabbix y tener credenciales para ingresar, puede ser un usuario y la password o un API Token (recomendado).

Zabbix API controla los permisos de la misma forma que lo hace el Frontend.

Breve introducción API de Zabbix

API Zabbix funcionamiento.



Mi computadora

Frontend

Requerimiento JSON - JSON especificando que requerimos

Respuesta JSON - Resultado del requerimiento

API Requerimiento JSON

Solicitud a Zabbix, información o modificaciones.

<pre>{ "jsonrpc": "2.0", "method": "Nombre de la función", "params": { parámetros de la función }, "auth": "Token de autorización", "id": 1 }</pre>	<p>Obligatorio y fijo</p> <p>Una acción a llevar adelante</p> <p>Puede ser un array</p> <p>API token o token de login</p> <p>Obligatorio pero no le damos uso</p>
---	---

API Requerimiento JSON

"method": "Objeto.Acción"

Objeto: “Conceptos Zabbix”

host
hostgroup
template
item...

Acción: Que queremos hacer

get
delete
create
update ...

No todos los objetos admiten todas las funciones, por ej, no se puede crear un evento, va a ser consecuencia del monitoreo.

API Requerimiento JSON

"params":{ parámetros de la función }

get

Si estamos pidiendo información, los parámetros van a determinar los filtros.

update, delete ...

Si estamos modificando información los parámetros van a ser los **objetos** con lo que vamos a trabajar

API Respuesta JSON

```
{  
  "jsonrpc": "2.0",  
  "result": { resultado de la acción },  
  "id": 1  
}
```



La respuesta viene en la propiedad result, su contenido depende de la función
get **objetos** solicitados
update, delete ... id de los **objetos** con los que trabajamos

Bajar la Appliance de Zabbix o disponer de una instalación



- Desde la [página de Zabbix](#), descargar .ovf y extraer el archivo

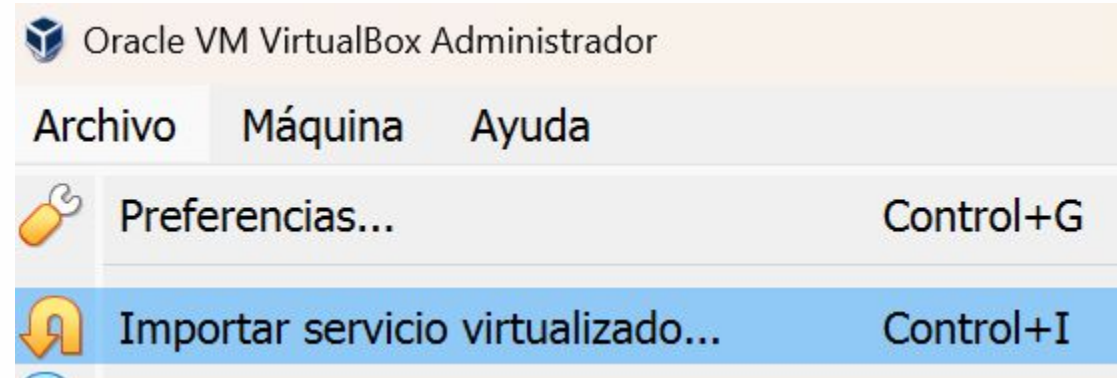
Zabbix Packages	Zabbix Cloud Images	Zabbix Containers	Zabbix Appliance	Zabbix Sources	Zabbix Agents
-----------------	---------------------	-------------------	-------------------------	----------------	---------------

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

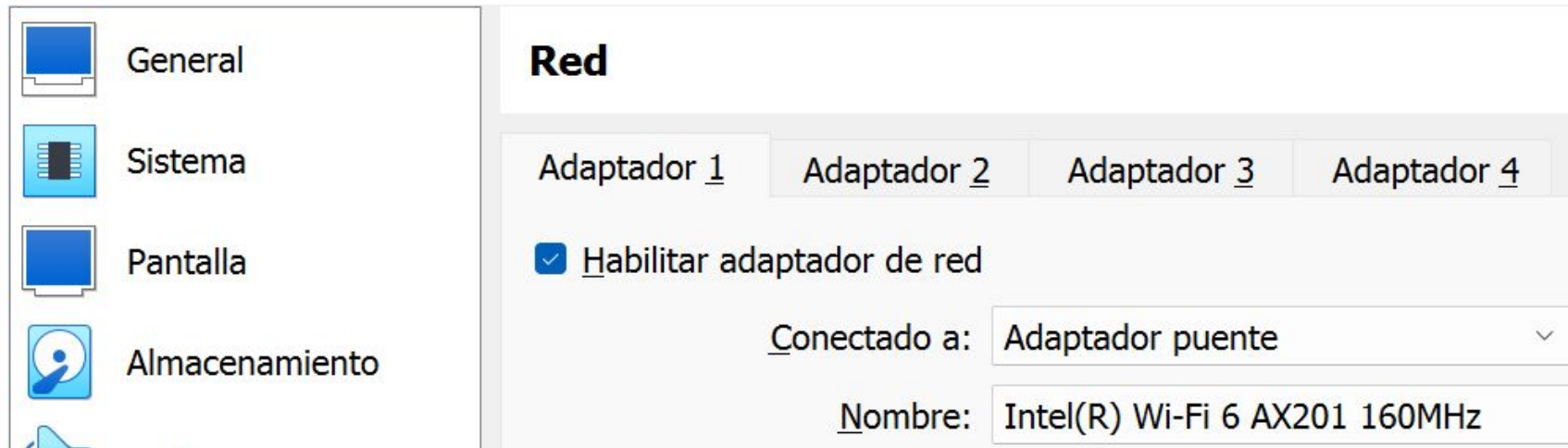
- Descomprimir el archivo

Bajar la Appliance de Zabbix o disponer de una instalación

- Importar servicio virtualizado en VirtualBox



- Configurar la Red de la VM a “Adaptador puente”



Bajar la Appliance de Zabbix o disponer de una instalación

1. Loggerarse en la VM

- root:zabbix
 - Buscar la IP
 - ip -a
- hostname -I



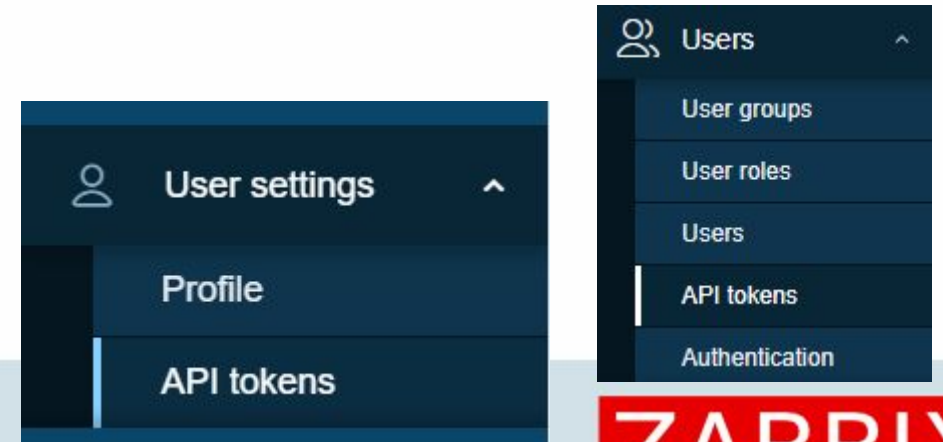
```
zabbix_appliance-7.0.0 [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
[root@appliance ~]# hostname -I
192.168.248.179 2800:ac:1:f78c:a00:27ff:fedd:99e
[root@appliance ~]# _
```

2. Loggearnse desde una terminal (WSL ?)

- ganamos facilidades como copiar y pegar, teclado etc
- ssh root@192.168.248.179

3. Entrar en el frontend de la máquina virtual

- Admin:zabbix
- Generar un API token (guardarlo)



Preparar el ambiente

1. Clonar el repositorio de CZAPI

a. Se recomienda actualizar WLS

`sudo apt update && sudo apt upgrade`

b. Ir a un directorio donde vamos a trabajar

i. `mkdir WebinarCustosAPI && cd WebinarCustosAPI`

c. `git clone` <https://github.com/CUSTOSMonitoring/CZAPI.git>

d. Se creó un directorio CZAPI

i. `cd CZAPI`

2. Preparar el ambiente

a. `. SetEnvironment`



el punto es para obligar a que este shell ejecute el cmd

Generar la conexión a la Appliance



1. ZAPI -C W<tab>

- a. va a generar un archivo para conectarnos a la Máquina virtual, copia de un ejemplo, que va a completar con el <Tab>

2. Completar las opciones

- a. Para esto se ingresa el nombre de la opción (o parte) o el número
- b. Ya hay cargadas opciones
 - i. 5) DEF_FUNC
 - ii. 16) format
 - iii. 17) libreria
- c. 12) URL
 - i. http://192.168.1.17 (la ip de la máquina virtual, o de nuestra instalación)
- d. 15) authToken El token generado en el frontend
- e. 1) Salir
 - i. Salvar el archivo, el enter va a dejar el mismo nombre, sobrescribimos el original

Hosts



1. ZAPI -c W<tab>
 - a. va a completar con el archivo de conexión del ejemplo
 - b. podemos generar nuestros propios archivos para conectarnos a otras URL y versiones
2. ZAPI -c/WebinarApi7.0.zapi Host<tab><tab>
 - a. Se van a mostrar las funciones posibles
3. ZAPI -c/WebinarApi7.0.zapi HostG<tab>
 - a. Se completa con HostGet
4. <Enter>
5. Muestra la lista de los hosts definidos en la instalación Zabbix
 - a. Los **códigos son traducidos** para que sean entendibles
 - i. esta traducción es configurable

Templates

1. ZAPI -c W<tab>
 - a. va a completar con el archivo de conexión del ejemplo
 - b. podemos generar nuestros propios archivos para conectarnos a otras URL y versiones
2. ZAPI -c/WebinarApi7.0.zapi Templa<tab><tab>
 - a. Se van a mostrar las funciones posibles
3. ZAPI -c /WebinarApi7.0.zapi TemplateG<tab>
 - a. Se completa con TemplateGet
4. <Enter>
5. Muestra la lista de los templates definidos en la instalación Zabbix
 - a. Los **códigos son traducidos** para que sean entendibles
 - b. esta traducción es configurable

Templates

1. Para evitar especificar el archivo de configuración cada vez
 - a. `export ZAPI_Conf=/PATH/AL/ARCH/CONF.zapi`
2. ZAPI TeplateGet
 - a. Observar las columnas que recupera
 - i. muchas no corresponden con un template
 1. Este tipo de errores, deseos etc pueden ser enviados a [Zabbix](#)
 - ii. puede que en listados específicos no nos interesen algunas

Sacar columnas del listado

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"  
  } ...  
]
```

Modificar el resultado

Formatear el resultado para levantarlo en una planilla de cálculo

1. Vamos a generar un cvs
 - a. **jq** tiene “formateadores”
 - i. `@csv` genera un “comma separated values”
 - ii. recibe vectores con los valores por columnas
2. ZAPI TemplateGet **output**=host,name,description,uuid,vendor_name,vendor_version | \
jq ' **.[]** '
 - a. “Desarmamos” el array, tenemos cada template pero no “en vector”
3. ZAPI TemplateGet **output**=host,name,description,uuid,vendor_name,vendor_version | \
jq ' **.[] | [.]** '
 - a. Tenemos vectores pero objetos dentro, necesitamos strings
4. ZAPI TemplateGet **output**=host,name,description,uuid,vendor_name,vendor_version | \
jq ' **.[] | [.[]]** '
 - a. Este si

Formatear el resultado para levantarlo en una planilla de cálculo

1. ZAPI TemplateGet output=host,name,description,uuid,vendor_name,vendor_version | \njq ' .[] | [.[]] | @csv'
 - a. genera con muchos escapes de comillas
2. ZAPI TemplateGet output=host,name,description,uuid,vendor_name,vendor_version | \njq -r ' .[] | [.[]] | @csv'
 - a. eliminadas las comillas extra
3. ZAPI TemplateGet output=host,name,description,uuid,vendor_name,vendor_version | \njq -r ' .[] | [.[]] | @csv' > **ReporteTemplates.csv**
4. Abrir el archivo en una “Hoja de cálculo” (libreoffice, google, excel)
 - a. Faltan los títulos
 - i. ZAPI ... | jq -r ' (.[1] | keys), .[] | [.[]] | @csv' > ReporteTemplates.csv
 - b. Le doy un formato lindo

Agregar información

1. ZAPI HostGet **selectParentTemplates**

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


Agregar información

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"  
    }, ...  
  ]  
}]
```

Agregar información

```
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  
' > ReporteHosts.csv
```

Para aprender más de jq
[Manual de jq](#)

Para sugerir nuevos webinars, cambios a los existentes ...
capacitaciones@custos.uy

Ingresen en la comunidad, de la colaboración nace la fuerza, puede ser [Telegram](#) en español,
[Zabbix users group](#) en inglés, entre otras

Preguntas



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



info@custos.uy



[/custos-monitoring](https://custos-monitoring)



WEBINAR

¡Muchas gracias!

ZABBIX 

WEBINAR '24

ZABBIX

