**🛠️ Registro masivo de hosts Windows en Zabbix vía API**

**1. 🎯 Objetivo**

Automatizar el alta de múltiples clientes Windows en el servidor Zabbix mediante su API JSON-RPC. Se usará un script en Bash que recibe una lista de hosts (nombre e IP) y los registra bajo un grupo y template predefinidos.

**2. 📋 Requisitos previos**

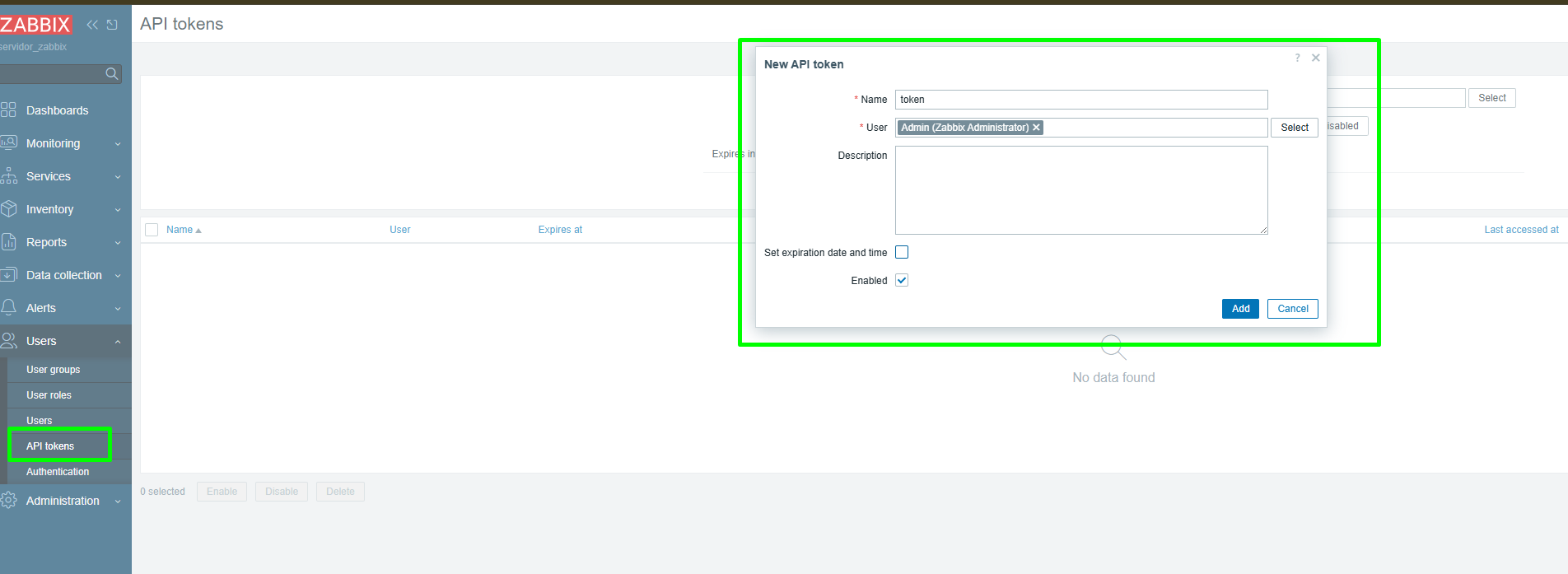
**2.1 En el servidor Linux desde donde se ejecuta el script:**

* Acceso a curl y jq:

sudo dnf install -y jq curl

**2.2 En Zabbix:**

* Tener disponible un **token de autenticación de API** generado para un usuario Zabbix con permisos suficientes.



* Conocer la URL del endpoint de la API JSON-RPC (por defecto: /zabbix/api\_jsonrpc.php).

**Con este comando de curl comprobamos que la API de Zabbix está accesible y operativa:**

curl --request POST --url 'http://23.96.90.111:8081/zabbix/api\_jsonrpc.php' --header 'Content-Type: application/json-rpc' --data '{"jsonrpc":"2.0","method":"apiinfo.version","params":{},"id":1}'

* Tener creado o identificado el nombre del grupo de hosts y del template deseado.

**3. 📌 Datos utilizados**

| **Elemento** | **Valor** |
| --- | --- |
| URL API Zabbix | http:// 23.96.90.111:8081/zabbix/api\_jsonrpc.php |
| Token API | 598a55e70bb72ea78d5fe1596b660cd439a224fa1035fa23fcc525ccec302a9c |
| Nombre del grupo | Windows hosts |
| Template usado | Windows by Zabbix agent |

**4. 📄 Estructura del script**

El script hace lo siguiente:

1. Comprueba si el grupo existe, si no lo crea.
2. Obtiene el ID del template.
3. Itera sobre una lista de nombre,ip.
4. Llama al método host.create por cada host.

4eae64b06986d220ebd4152aa3a8533a37177718481dad277fdfe0bb6c037b57

1. **📜 Código del script**

**registrar\_hosts.sh:**

#!/bin/bash

ZABBIX\_URL="http://23.96.90.111:8081/zabbix/api\_jsonrpc.php"

TOKEN="token"

GROUP\_NAME="Windows hosts"

TEMPLATE\_NAME="Windows by Zabbix agent"

HEADER\_AUTH="Authorization: Bearer $TOKEN"

HEADER\_JSON="Content-Type: application/json"

# Lista de hosts en formato: "nombre,ip"

HOSTS=(

"cliente-win-001,104.41.149.24"

"cliente-win-002,52.170.57.21"

"cliente-win-003,52.170.57.22"

# Añadir hasta 1000...

)

# Obtener o crear el grupo

GROUP\_ID=$(curl -s -X POST "$ZABBIX\_URL" -H "$HEADER\_JSON" -H "$HEADER\_AUTH" -d "{

\"jsonrpc\": \"2.0\",

\"method\": \"hostgroup.get\",

\"params\": { \"output\": [\"groupid\"], \"filter\": { \"name\": [\"$GROUP\_NAME\"] } },

\"id\": 1

}" | jq -r '.result[0].groupid')

if [[ -z "$GROUP\_ID" || "$GROUP\_ID" == "null" ]]; then

GROUP\_ID=$(curl -s -X POST "$ZABBIX\_URL" -H "$HEADER\_JSON" -H "$HEADER\_AUTH" -d "{

\"jsonrpc\": \"2.0\",

\"method\": \"hostgroup.create\",

\"params\": { \"name\": \"$GROUP\_NAME\" },

\"id\": 1

}" | jq -r '.result.groupids[0]')

echo "🆕 Grupo '$GROUP\_NAME' creado con ID $GROUP\_ID"

else

echo "✅ Grupo encontrado: $GROUP\_NAME → $GROUP\_ID"

fi

# Obtener el templateid

TEMPLATE\_ID=$(curl -s -X POST "$ZABBIX\_URL" -H "$HEADER\_JSON" -H "$HEADER\_AUTH" -d "{

\"jsonrpc\": \"2.0\",

\"method\": \"template.get\",

\"params\": { \"output\": [\"templateid\"], \"filter\": { \"name\": [\"$TEMPLATE\_NAME\"] } },

\"id\": 1

}" | jq -r '.result[0].templateid')

if [[ -z "$TEMPLATE\_ID" || "$TEMPLATE\_ID" == "null" ]]; then

echo "❌ Template '$TEMPLATE\_NAME' no encontrado."

exit 1

fi

echo "✅ Template encontrado: $TEMPLATE\_NAME → $TEMPLATE\_ID"

echo

# Crear los hosts

for entry in "${HOSTS[@]}"; do

nombre=$(echo "$entry" | cut -d',' -f1)

ip=$(echo "$entry" | cut -d',' -f2)

echo "🖥️ Registrando $nombre ($ip)..."

RESPONSE=$(curl -s -X POST "$ZABBIX\_URL" -H "$HEADER\_JSON" -H "$HEADER\_AUTH" -d "{

\"jsonrpc\": \"2.0\",

\"method\": \"host.create\",

\"params\": {

\"host\": \"$nombre\",

\"interfaces\": [{

\"type\": 1,

\"main\": 1,

\"useip\": 1,

\"ip\": \"$ip\",

\"dns\": \"\",

\"port\": \"10050\"

}],

\"groups\": [{\"groupid\": \"$GROUP\_ID\"}],

\"templates\": [{\"templateid\": \"$TEMPLATE\_ID\"}]

},

\"id\": 1

}")

if echo "$RESPONSE" | grep -q "hostids"; then

echo "✅ Host $nombre creado correctamente."

else

echo "❌ Error con $nombre:"

echo "$RESPONSE" | jq

fi

echo "-------------------------------------"

done

**6. ▶️ Ejecución del script**

1. Asigna permisos de ejecución:

chmod +x registrar\_hosts.sh

1. Ejecuta el script:

./registrar\_hosts.sh