Zabbix MCP Server with Gemini CLI

Quick Guide (CLI)

1. Installation

To use the Zabbix MCP Server you need the following:

- a stable internet connection,
- Docker,
- local Zabbix server,
- local Zabbix MCP server,
- Gemini CLI.

If you don't have the Gemini CLI installed, refer back to my previous guide about Installing Gemini CLI on your device.

First to be able to do anything you have to install Docker on your device. Here's a tutorial on how to do it:

ON LINUX:

> *sudo apt update*

> *sudo apt install -y docker.io docker-compose*

> *sudo systemctl enable --now docker*

ON MAC:

> Go to this website: DOCKER INSTALL

> Download the correct version of the software and install it on your device (REMEMBER TO CHOOSE THE CORRECT CPU!)

> Run the Docker Desktop Application

ON WINDOWS:

> Go to this website: DOCKER INSTALL

> Download the correct version of the software and install it on your device

> Reboot your device

> Run the Docker Desktop Application

To check if Docker installed correctly on your device type the following commands into your terminal:
> docker -v
> docker-compose -v

If you have installed Docker correctly the next step is to start the Zabbix local server on your device.

2. Configuration

To do that you need to create a directory (preferably named along the lines of “zabbix_server”) and create a **docker-compose.yml** file inside that directory.
That file allows you to use Docker to create a Docker container, download needed files and run the server using one command.

Here is the docker-compose.yml file:

```
version: '3.5'                          # Docker Compose file format
version

services:                               # Define the services
(containers) to be created

  mysql:                                # MySQL/MariaDB service
    image: mariadb:10.5                 # Use MariaDB version 10.5
image
    container_name: zbx-mysql           # Name of the container
    environment:                        # Set environment variables
for MariaDB
      MYSQL_ROOT_PASSWORD: root_pass    # Root password for the
database
      MYSQL_DATABASE: zabbix            # Create a database named
'zabbix'
      MYSQL_USER: zabbix                # Create a user named 'zabbix'
      MYSQL_PASSWORD: zabbix_pass       # Password for user 'zabbix'
    volumes:
```

```
      - zabbix-db:/var/lib/mysql          # Persist database data using
named volume
    restart: unless-stopped               # Restart unless the container
is manually stopped

  zabbix-server:                          # Zabbix server service
    image: zabbix/zabbix-server-mysql:alpine-7.0-latest    # Use
lightweight Alpine image of Zabbix server (MySQL variant)
    container_name: zbx-server            # Name of the container
    environment:
      DB_SERVER_HOST: mysql               # Hostname of the database
(name of the MySQL service)
      MYSQL_DATABASE: zabbix              # Database name to connect to
      MYSQL_USER: zabbix                  # Database username
      MYSQL_PASSWORD: zabbix_pass         # Database password
    depends_on:
      - mysql                             # Ensure this service starts
after the MySQL service
    ports:
      - "10051:10051"                     # Expose Zabbix server port
    restart: unless-stopped               # Auto-restart policy

  zabbix-web:                             # Zabbix web frontend service
    image: zabbix/zabbix-web-nginx-mysql:alpine-7.0-latest # Use Alpine
image with Nginx + MySQL support
    container_name: zbx-web               # Name of the container
    environment:
      DB_SERVER_HOST: mysql               # Database host
      MYSQL_DATABASE: zabbix              # Database name
      MYSQL_USER: zabbix                  # Database user
      MYSQL_PASSWORD: zabbix_pass         # Database password
      ZBX_SERVER_HOST: zabbix-server      # Hostname of the Zabbix
server
      PHP_TZ: Asia/Ho_Chi_Minh            # Timezone for the PHP
frontend
    ports:
      - "8080:8080"                       # Expose web interface on port
8080
    depends_on:
      - zabbix-server                     # Ensure Zabbix server is up
before starting this service
    restart: unless-stopped               # Auto-restart if stopped
unexpectedly
```

```
volumes:
  zabbix-db:                              # Define a named volume to
persist MySQL data
```

After creating that file open the directory of that file in terminal and type in the following command:

ON WINDOWS AND MAC:

> docker compose up -d

ON LINUX:

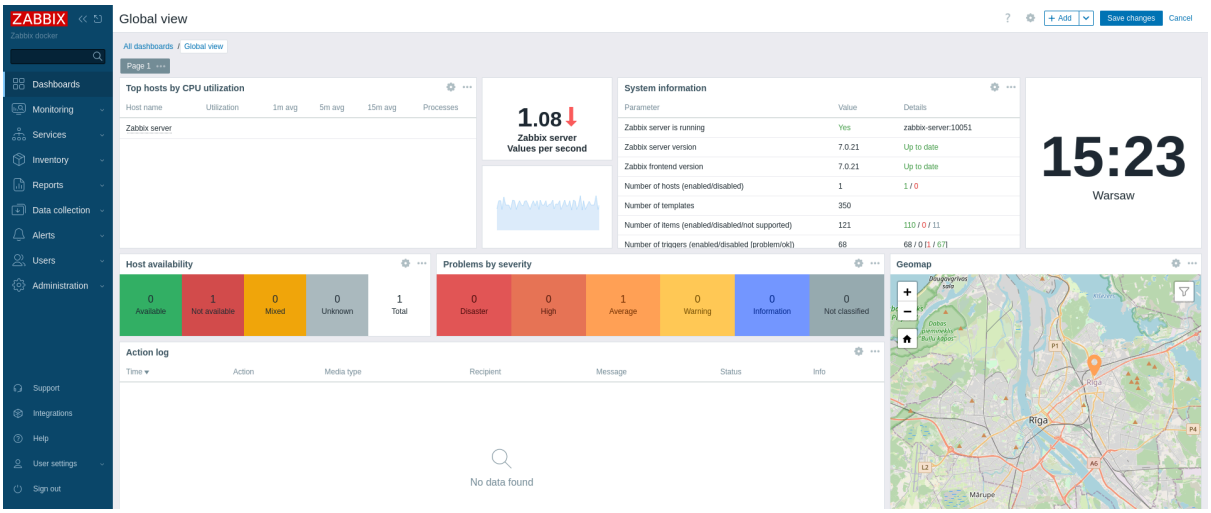
> sudo docker compose up -d

This command will create a Docker container, pull the server image and run it on your device. Using the -d flag allows us to free the terminal immediately after running the command. Now you need to wait about 5 - 10 minutes so the server fully starts. You can check its progress by going to the ***http://localhost:8080*** link.

You can also optionally check the logs by using the following command:

> docker compose logs -f

If everything went according to plan, when visiting ***http://localhost:8080*** you should see this on your screen.



3. Zabbix MCP Server Installation and Configuration

After you successfully install and configure the Zabbix local server now we can install and configure the MCP server.

First you need to clone the files from the git repository using the following command:

> *git clone* https://github.com/mpeirone/zabbix-mcp-server.git

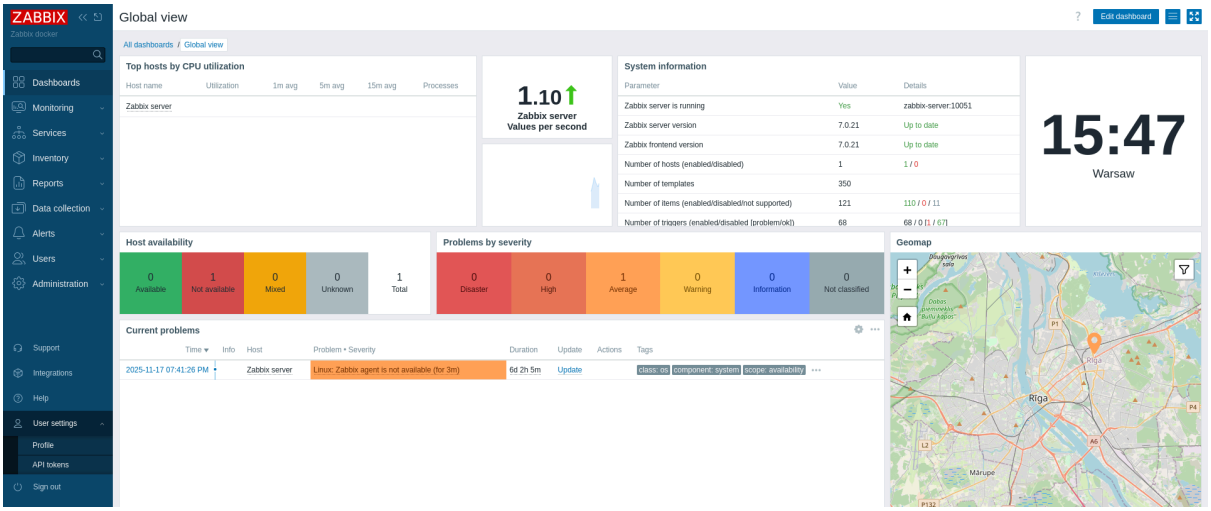
Next you need to insert the following JSON code into the settings.json file (you can find the file in the following directory *~/.gemini*):

```
"mcpServers": {
    "zabbix": {
      "command": "uv",
      "args": [
        "run",
        "--directory",
        "/path/to/zabbix-mcp-server",
        "python",
        "src/zabbix_mcp_server.py"
      ],
      "env": {
        "ZABBIX_URL": "http://localhost:8080",
        "ZABBIX_TOKEN": "<your_api_token>",
```

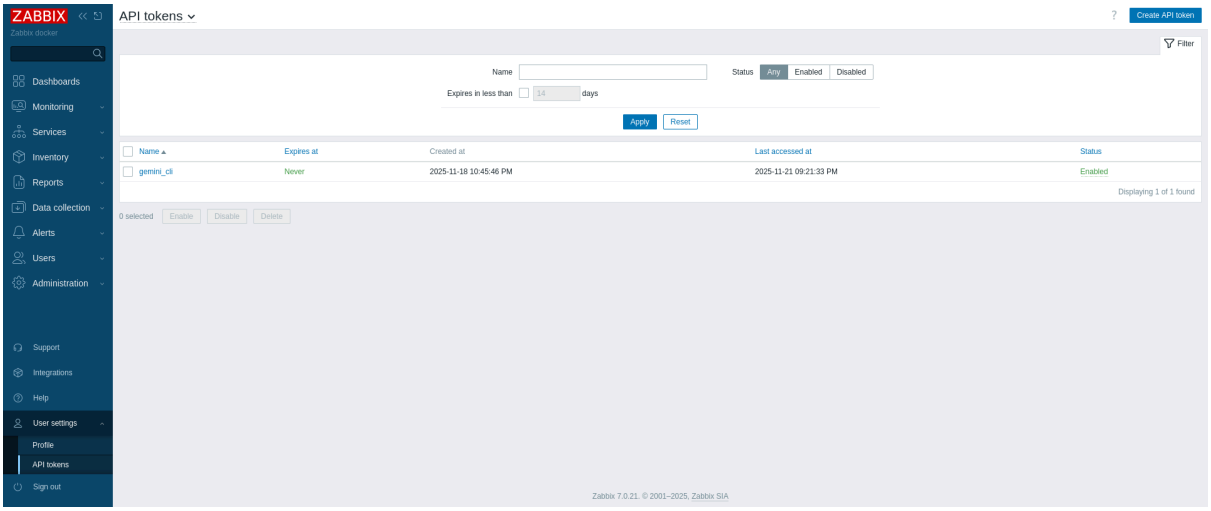
```
            "READ_ONLY": "true"
        }
    }
}
```

To authenticate the MCP server you need an API token. To get it follow the following steps:

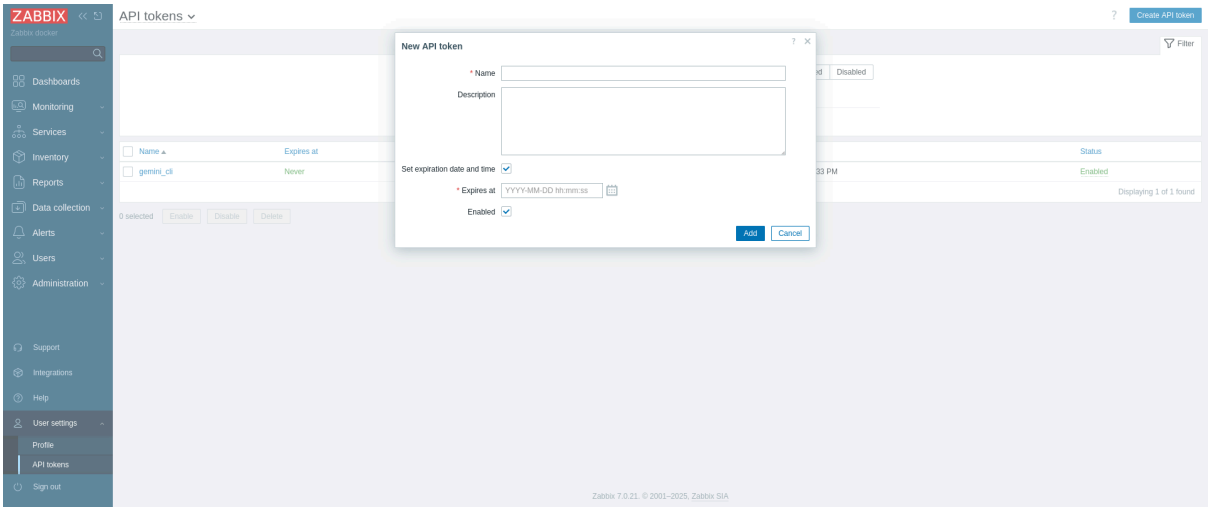
- On the dashboard go to API Tokens



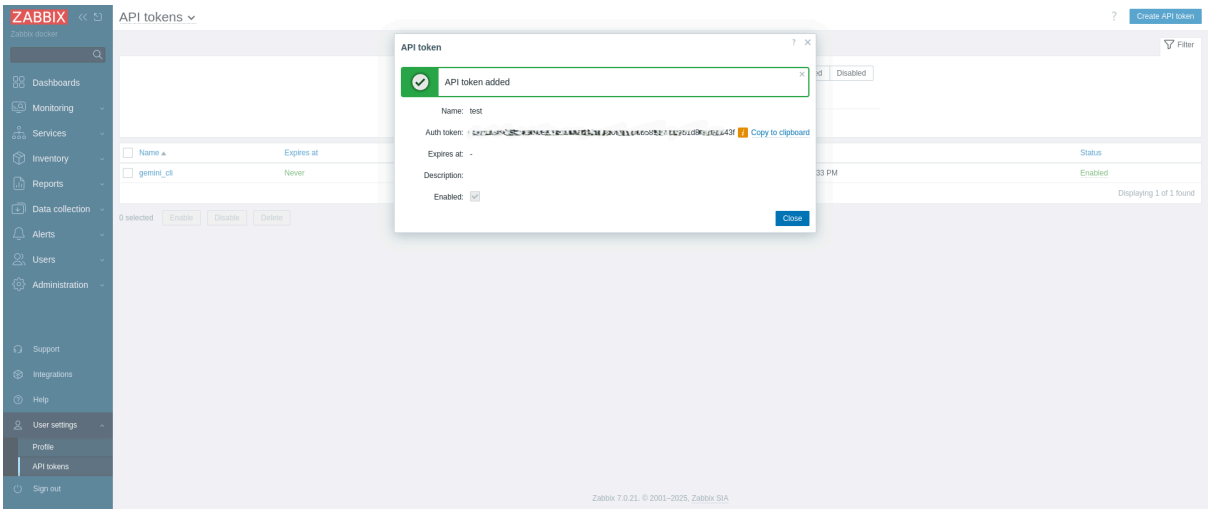
- In API Tokens click “Create API Token”



- Here fill out the form and click “Add”



- After creating the API token copy the auth token and paste it into the settings.json file under “ZABBIX_TOKEN”



⚠**REMEMBER TO FILL OUT THESE VARIABLES IN THE SETTINGS.JSON FILE CORRECTLY:**

- **ZABBIX_TOKEN**
- **ZABBIX_URL**

AND THE ARGUMENTS CONTAINING YOUR PATH WHERE YOU COPIED THE GIT REPO.⚠

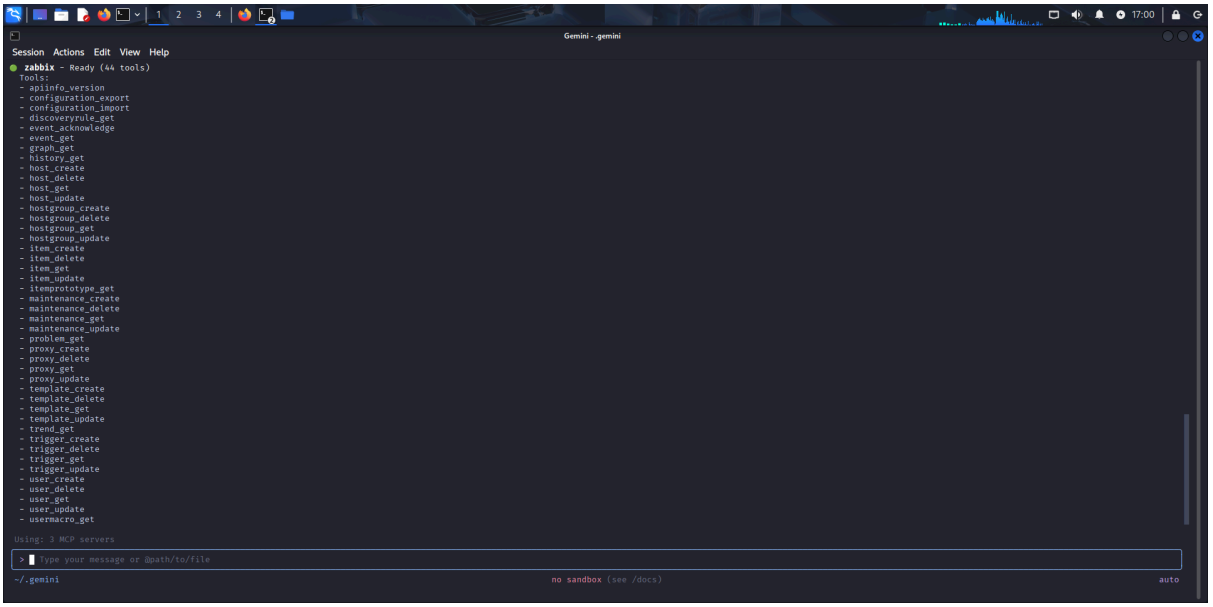
Now after doing all previous steps open the directory in which you copied the github repo and use the following command to run the MCP Server:
and type in the following command:

ON WINDOWS AND MAC:
> docker compose up -d

ON LINUX:
> sudo docker compose up -d

After finishing all previous steps you can run Gemini in your terminal and check if the MCP Server installed and authenticated correctly by using the following command
> /mcp list

If everything went according to plan you should see this in your Gemini CLI:



CONGRATULATIONS! NOW YOU CAN USE THE ZABBIX MCP SERVER TO COMMUNICATE WITH YOUR ZABBIX SERVER!