

AI with a web browser can simplify everyday Zabbix tasks and inspire new ways of working

K.Inaba from Japan

I'm K.Inaba aka nabaua.

I've been working IT for about 30 years.

Of course we are using Zabbix and monitoring them publicly and privately

Agenda

1. The scripts of Zabbix_sender by Google Browser AI

You can do that easily

I'll show you using by Google Browser AI.

2. The scripts of Zabbix trigger by Google Browser AI

You can do that easily

I'll show you using by Google Browser AI.

3. The more little requirements the script are more elaborated

You can do that easily

I'll show you using by Google Browser AI.

Recently, making code is not using manually but doing with AI in a high quality machine or high GPU like a LLM or super machine.

Doing them, I think it's expensive.

So, if you want to have some scripts of zabbix_sender or tinger which are not heavy, I think we don't need to use them.

In this topic, I'll talk about these scripts to be made automatically with AI mode in Google browser.

Listen please.

1. The scripts of Zabbix_sender by Google Browser AI

The requirements we text is like this in a google browser.

Key word are very important.

[Write the templates of the code for monitoring the A server by Zabbix sender]

When we input that, and then



🔍 Write the templates of the code for monitoring the A server by Zabbix sender|

🔄 AI Mode

EX

Bash Script like a Template on a browser immediately.

Use this template for quick automations, cron jobs, or lightweight system monitoring.bash

```
#!/bin/bash
```

```
# Configuration
```

```
ZABBIX_SERVER="192.168.1.100" # Zabbix Server IP or DNS
```

```
HOST_NAME="Target-Server-01" # Must match the Host name in Zabbix precisely
```

```
ITEM_KEY="server.metric.status" # Must match the Item Key in Zabbix
```

```
PORT="10051" # Default Zabbix trapper port
```

```
# Calculate or fetch your metric value
```

```
METRIC_VALUE=$(vmstat 1 2 | tail -1 | awk '{print $15}') # Example: Getting idle CPU
```

```
# Send the data to Zabbix
```

```
zabbix_sender -z "$ZABBIX_SERVER" -p "$PORT" -s "$HOST_NAME" -k "$ITEM_KEY" -o "$METRIC_VALUE"
```

```
# Check exit status
```

```
if [ $? -eq 0 ]; then
```

```
    echo "Data sent successfully."
```

```
else
```

```
    echo "Failed to send data to Zabbix."
```

```
fi
```

Or Python script

Use this template for quick automations, cron jobs, or lightweight system monitoring.python

```
import subprocess
from zabbix_utils import Sender
# Configuration
ZABBIX_SERVER = "192.168.1.100"
HOST_NAME = "My Target Server"
def get_metric():
    # Replace this with your actual monitoring logic
    return 85.5
def main():
    metric_value = get_metric()

    # Initialize the sender
    sender = Sender(server=ZABBIX_SERVER)

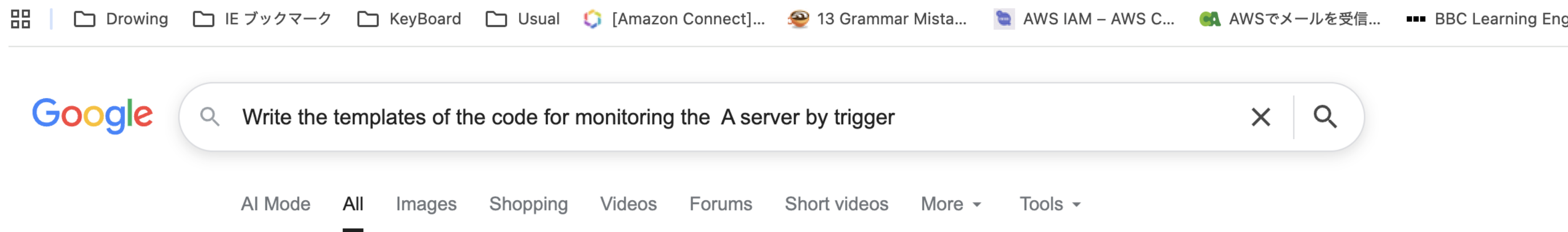
    # Send the data (Host, Key, Value)
    response = sender.send_value(HOST_NAME, "custom.server.metric", metric_value)

    # Check response
    if response.failed == 0:
        print(f"Successfully sent data: {response}")
    else:
        print(f"Failed to send data: {response}")
if __name__ == "__main__":
    main()
```

Of course
We can customize them.

2.The scripts of Zabbix trigger by Google Browser AI

The requirements we text is like this in a google browser.
[Write the templates of the code for monitoring the triger]
When we input that, and then



The code is made from Google browser all of sudden when you enter key
The content of the trigger is randomly written.
Python
This case is the trigger limit 80 % CPU

```
import psutil
import time
# Set the trigger limit (80%)
CPU_LIMIT = 80.0
def check_server():
    # Get current CPU usage percent
    cpu_usage = psutil.cpu_percent(interval=1)

    if cpu_usage > CPU_LIMIT:
        send_alert(cpu_usage)
    else:
        print(f"Server is healthy. CPU: {cpu_usage}%")
def send_alert(cpu_usage):
    # Add your alert logic here (e.g., webhook, email, or log file)
    print(f"ALERT! High CPU usage detected: {cpu_usage}%")
if __name__ == "__main__":
    print("Starting server monitor...")
    while True:
        check_server()
        # Wait 60 seconds before next check
        time.sleep(60)
```

3.The little more requirements the script are the little more sophisticated scripts are created.

[write the templates of the code for monitoring the server by zabbix_sender whose the content is over 90 % CPU and , finally shut down]



write the templates of the code for monitoring the a server by zabbix_sender whose content is over 90



AI Mode

All

Images

Shopping

Videos

Forums

Short videos

More ▾

Tools ▾

The code we require are made immediately.

```
#!/bin/bash
# Configuration
ZABBIX_SERVER="192.168.1.100"    # Replace with your Zabbix Server IP
HOST_NAME="Target-Server-Name"  # Replace with the exact Host Name in Zabbix
ITEM_KEY="custom.cpu.usage"     # The Zabbix item key
# Calculate CPU usage (100 - idle time)
CPU_IDLE=$(top -bn1 | grep "Cpu(s)" | sed "s/.*, *\[0-9.\]*\)%* id.*\1/")
CPU_USAGE=$(echo "100 - $CPU_IDLE" | bc)
# Send the metrics to Zabbix using zabbix_sender
zabbix_sender -z "$ZABBIX_SERVER" -s "$HOST_NAME" -k "$ITEM_KEY" -o "$CPU_USAGE"
# Check if CPU usage is over 90%
TRIGGER_LIMIT=90.0
IS_OVER=$(echo "$CPU_USAGE > $TRIGGER_LIMIT" | bc -l)
if [ "$IS_OVER" -eq 1 ]; then
    echo "Warning: CPU usage is at ${CPU_USAGE}%. Initiating system shutdown..."

    # Optional: Send a final critical alert to Zabbix before going down
    zabbix_sender -z "$ZABBIX_SERVER" -s "$HOST_NAME" -k "custom.server.status" -o "Shutting down due to high CPU"

    # Execute shutdown (requires script to run as root or with sudo permissions)
    sudo /sbin/shutdown -h now
fi
```

Yes.

Finally that monitoring system will cause it to shut down immediately.

I believe that the more requirements you list, the more sophisticated the resulting system will be.

What is truly needed is not the act of writing itself, but the requirements. I think that is what we need going forward. They will inspire us.

Thank you.