windows-monitoring-system

Monitoring on-premises servers is essential to ensure their performance, detect anomalies, and enhance security. This project leverages Python's psutil module to achieve detailed monitoring with a client-database architecture. Metrics are collected from on-premises servers, stored in a PostgreSQL database, and visualized on a Grafana dashboard.

Author : Shriniwas Kulkarni

- **PCCOE 2026 BTech CSE(AIML)**
- **Email:** kshriniwas180205@gmail.com
- **Phone:** +91 [8999883480]
- **GitHub:** github.com/Shriniwas18K

Features

- Real-time monitoring of server performance metrics.
- Anomaly detection and security insights.
- Seamless integration with Grafana for interactive data visualization.
- Modular design for customization to different operating systems.

Architecture Overview

The system uses a client-database architecture:

- Clients: On-premises servers that send performance metrics to the PostgreSQL database.
- Database: PostgreSQL is provisioned on the cloud to store metrics.
- Visualization: Metrics are visualized on a Grafana dashboard.

Setup and Usage

Follow the steps below to set up and use the system:

1. Install Python and Dependencies:

- i) Install Python on all on-premises Windows machines.

- ii) Install the psutil package: pip install psutil.

2. Provision PostgreSQL Database:

- i) Provision a PostgreSQL database on the cloud and note the connection string.

3. Database Setup:

- i) Download and run setup.py on any machine to create the required tables in the PostgreSQL database.

4. Configure Environment:

- i) Save the PostgreSQL connection URL as an environment variable on on-premises servers.

5. Deploy Client Script:

- i) Download the client.py script from the repository.
- ii) Run client.py on all on-premises servers to collect and send metrics to the database.

6. Visualize with Grafana:

- i) Connect your Grafana instance to the PostgreSQL database.
- ii) Create dashboards to visualize server metrics.

7. Data Retention Policy:

- i) By default, the client script deletes all records from PostgreSQL every day at midnight (00:00). You can modify this behavior in client.py.

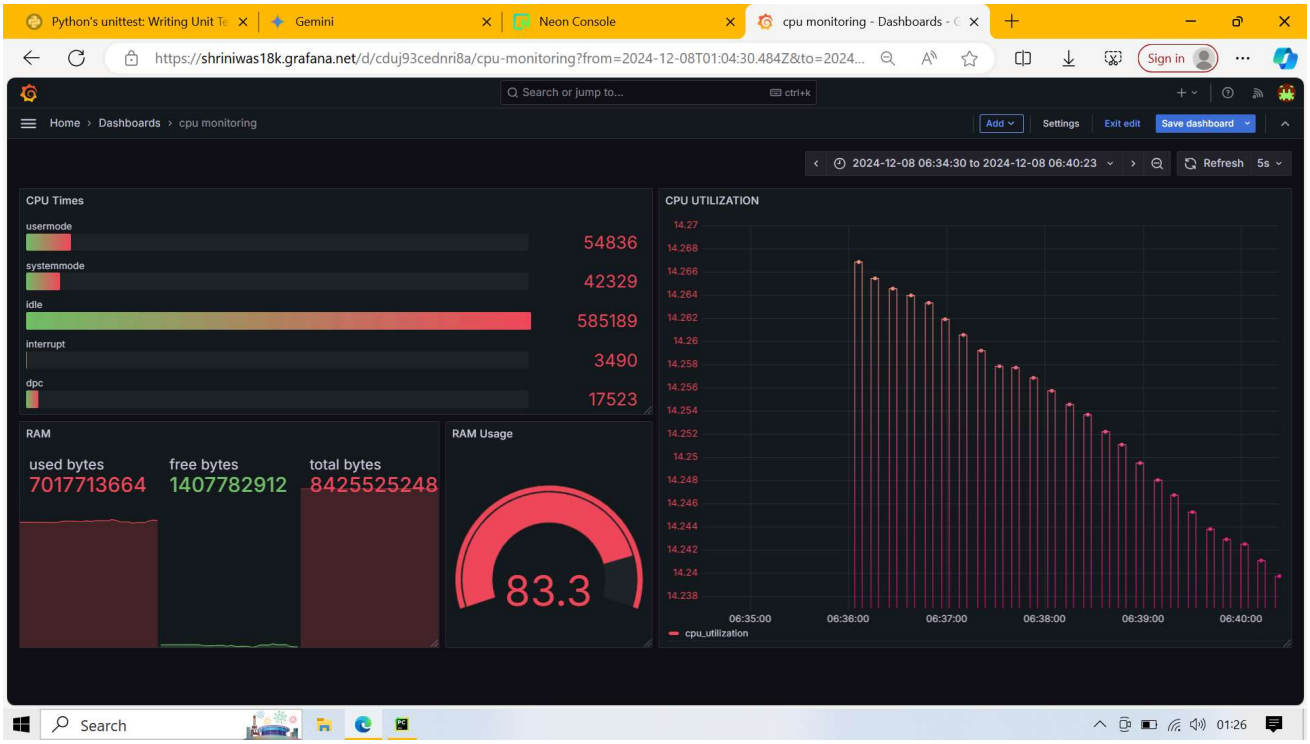
Customization

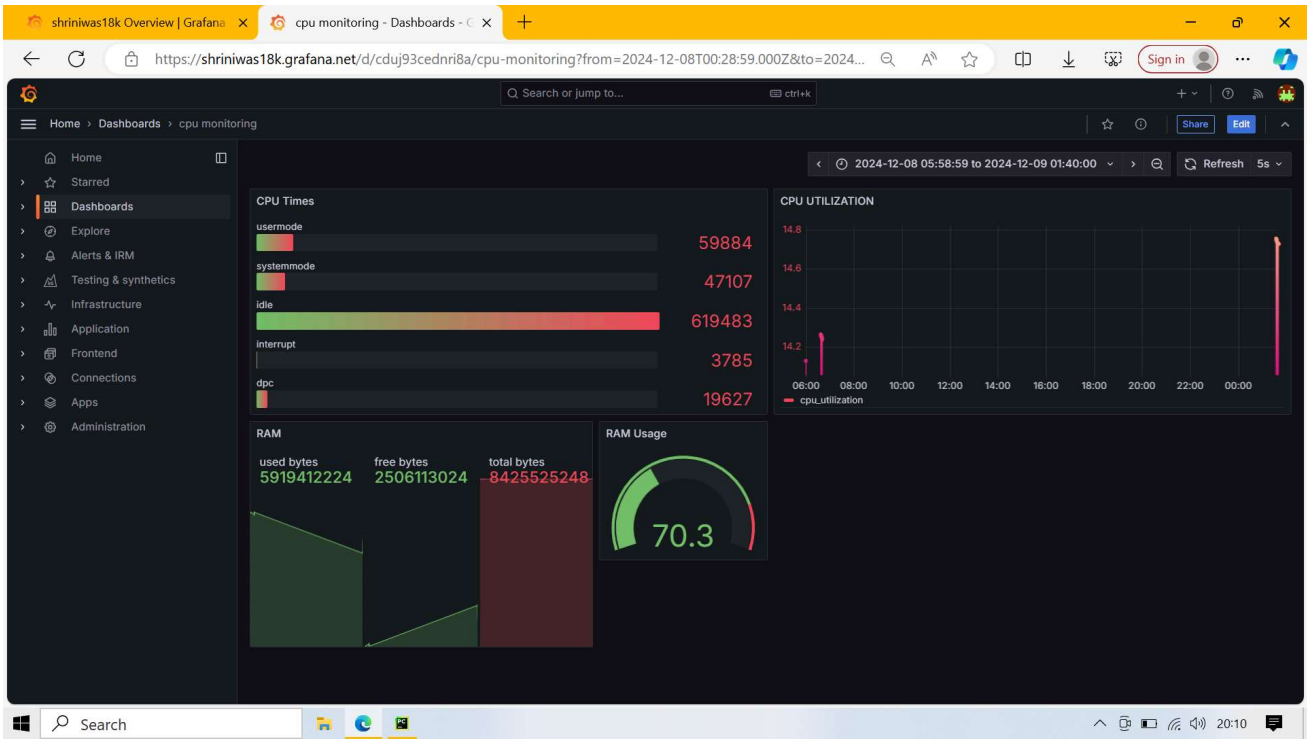
Refactor the code as needed to support specific operating systems for your on-premises servers.

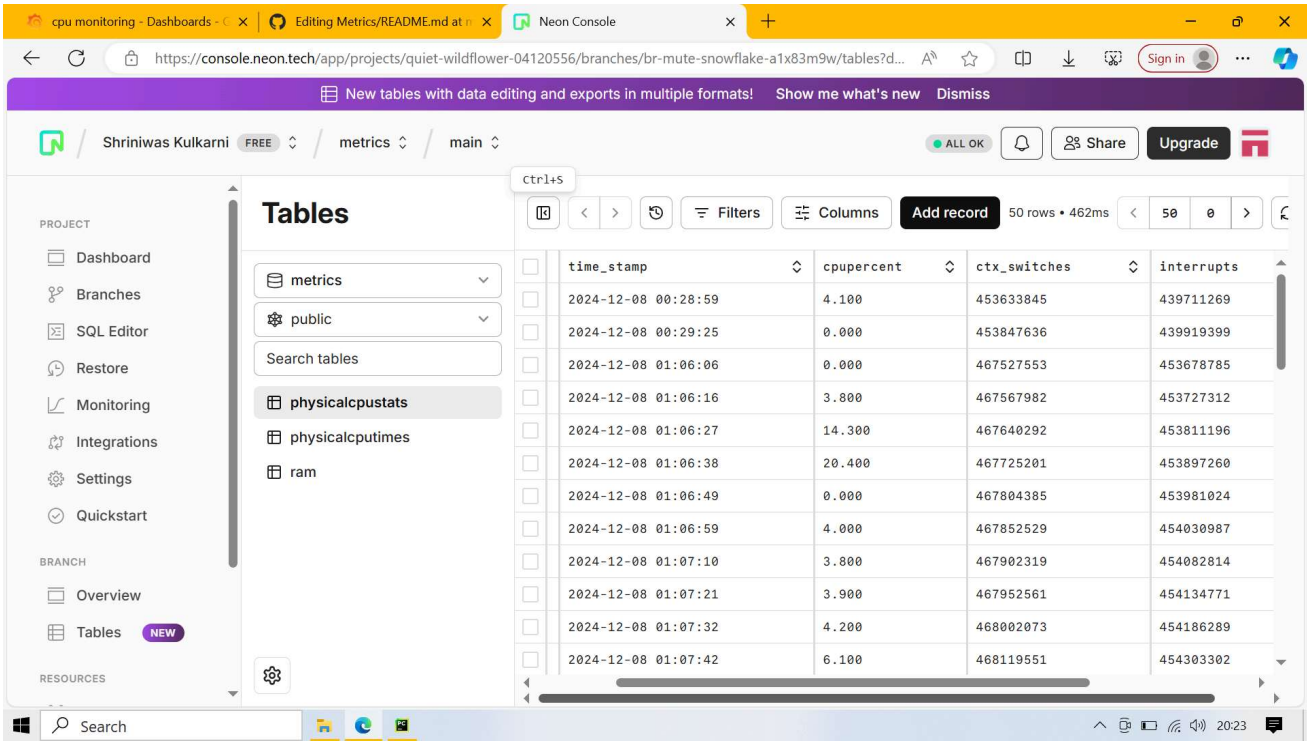
Testing

The codebase has been tested using unittest and doctest.

Screenshots


Python's unittest: Writing Unit Te
Gemini
Neon Console
cpu monitoring - Dashboards -
https://shriniwas18k.grafana.net/d/cduj93cednri8a/cpu-monitoring?from=2024-12-08T01:04:30.484Z&to=2024...
Sign in
Search or jump to...
ctrl+k
Home › Dashboards › cpu monitoring
Add
Settings
Exit edit
Save dashboard
2024-12-08 06:34:30 to 2024-12-08 06:40:23
Refresh
5s
CPU Times
usermode
54836
systemmode
42329
idle
585189
interrupt
3490
dpc
17523
RAM
used bytes
7017713664
free bytes
1407782912
total bytes
8425525248
RAM Usage
83.3
CPU UTILIZATION
14.27
14.268
14.266
14.264
14.262
14.26
14.258
14.256
14.254
14.252
14.25
14.248
14.246
14.244
14.242
14.24
14.238
06:35:00
06:36:00
06:37:00
06:38:00
06:39:00
06:40:00
cpu_utilization
Search
01:26


shriniwas18k Overview | Grafana
cpu monitoring - Dashboards -
https://shriniwas18k.grafana.net/d/cduj93cednri8a/cpu-monitoring?from=2024-12-08T00:28:59.000Z&to=2024...
Sign in
Search or jump to...
ctrl+k
Home › Dashboards › cpu monitoring
Share
Edit
Home
Starred
Dashboards
Explore
Alerts & IRM
Testing & synthetics
Infrastructure
Application
Frontend
Connections
Apps
Administration
2024-12-08 05:58:59 to 2024-12-09 01:40:00
Refresh
5s
CPU Times
usermode
59884
systemmode
47107
idle
619483
interrupt
3785
dpc
19627
RAM
used bytes
5919412224
free bytes
2506113024
total bytes
8425525248
RAM Usage
70.3
CPU UTILIZATION
14.8
14.6
14.4
14.2
06:00
08:00
10:00
12:00
14:00
16:00
18:00
20:00
22:00
00:00
cpu_utilization
Search
20:10


cpu monitoring - Dashboards -
Editing Metrics/README.md at
Neon Console
https://console.neon.tech/app/projects/quiet-wildflower-04120556/branches/br-mute-snowflake-a1x83m9w/tables?d...
Sign in
New tables with data editing and exports in multiple formats!
Show me what's new
Dismiss
Shriniwas Kulkarni
FREE
metrics
main
ALL OK
Share
Upgrade
PROJECT
Dashboard
Branches
SQL Editor
Restore
Monitoring
Integrations
Settings
Quickstart
BRANCH
Overview
Tables
NEW
RESOURCES
Tables
Ctrl+S
metrics
public
Search tables
physicalcpustats
physicalcputimes
ram
Filters
Columns
Add record
50 rows • 462ms
50
0
time_stamp
cpupercent
ctx_switches
interrupts
2024-12-08 00:28:59 4.100 453633845 439711269
2024-12-08 00:29:25 0.000 453847636 439919399
2024-12-08 01:06:06 0.000 467527553 453678785
2024-12-08 01:06:16 3.800 467567982 453727312
2024-12-08 01:06:27 14.300 467640292 453811196
2024-12-08 01:06:38 20.400 467725201 453897260
2024-12-08 01:06:49 0.000 467804385 453981024
2024-12-08 01:06:59 4.000 467852529 454030987
2024-12-08 01:07:10 3.800 467902319 454082814
2024-12-08 01:07:21 3.900 467952561 454134771
2024-12-08 01:07:32 4.200 468002073 454186289
2024-12-08 01:07:42 6.100 468119551 454303302
Search
20:23