Usage Steps: mpstat & pidstat parser tool

Table of Contents

[mpstat\_csv.py (Collect CPU Usage Data and Save to CSV) 1](#_Toc193803289)

[Overview 1](#_Toc193803290)

[Pre-requisites 2](#_Toc193803291)

[How to Run 2](#_Toc193803292)

[mpstat\_plot.py (Generate CPU Usage Reports & Graphs) 2](#_Toc193803293)

[Overview 2](#_Toc193803294)

[Pre-requisites 2](#_Toc193803295)

[How to Run 2](#_Toc193803296)

[Example Runs: 3](#_Toc193803297)

[CSV generation script 3](#_Toc193803298)

[Plotting script 3](#_Toc193803299)

[pidstat\_csv.py 4](#_Toc193803300)

[How to run? 4](#_Toc193803301)

[pidstat\_plot.py 4](#_Toc193803302)

[How to run? 4](#_Toc193803303)

The two scripts **mpstat\_csv.py** and **mpstat\_plot.py** work together to monitor CPU usage and generate visual reports based on the collected data.

**pidstat\_csv.py** gives CPU usage based on the process ID passed as input.

# mpstat\_csv.py (Collect CPU Usage Data and Save to CSV)

## Overview

This script collects CPU usage statistics using the **mpstat** command and saves the output to a CSV file.

It allows the user to specify which CPU cores to monitor, the sampling interval, the number of samples, the output directory for the CSV file and the filename for the output CSV.

## Pre-requisites

git clone https://github.com/GayathriNarayana19/mpstat\_pidstat\_parser.git

sudo apt install sysstat

## How to Run

Execute the script using:

**python3 mpstat\_csv.py**

# mpstat\_plot.py (Generate CPU Usage Reports & Graphs)

## Overview

This script analyzes and visualizes CPU usage data stored in CSV files. It generates bar chart comparisons for different CPU performance metrics and saves them as PDF reports.

## Pre-requisites

pip install matplotlib

pip install seaborn

## How to Run

**python3 mpstat\_plot.py**

# Example Runs:

### CSV generation script

A screen shot of a computer

AI-generated content may be incorrect.

If user doesn’t give any inputs and press enter, it will take the default inputs and create CSVs with unique names like cpu\_usage.csv, cpu\_usage1.csv, cpu\_usage2.csv and so on. . without overwriting.

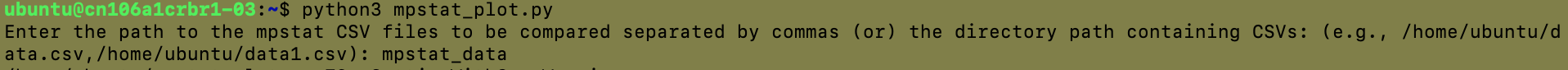
**A screen shot of a computer program

AI-generated content may be incorrect.**

### Plotting script

Choose any one method

##### Method 1: Just give dir name that contains CSV files generated from previous run like in the below screenshot.



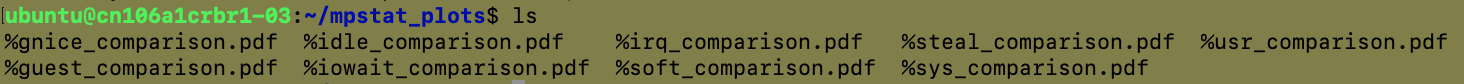
##### Method 2: Give the csv files along with path separated by commas

python3 mpstat\_plot.py

Enter the path to the mpstat CSV files to be compared separated by commas (or) the directory path containing CSVs: (e.g., /home/ubuntu/data.csv,/home/ubuntu/data1.csv): **/home/ubuntu/mpstat\_data/cpu\_usage1.csv, /home/ubuntu/mpstat\_data/cpu\_usage1.csv**

Output dir mpstat\_plots will be created and it will have all the outputs.





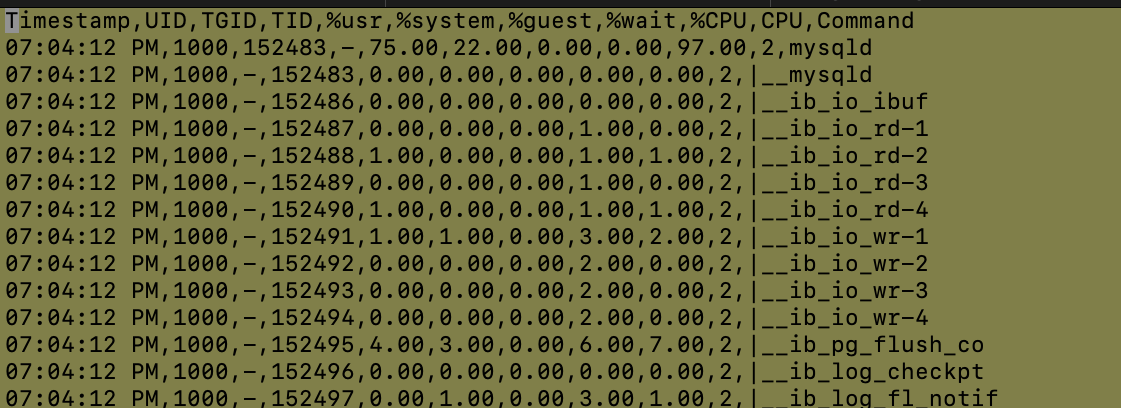
# pidstat\_csv.py

## How to run?

A screenshot of a computer code

AI-generated content may be incorrect.

##### CSV format



# pidstat\_plot.py

## How to run?

python3 pidstat\_plot\_v2.py

Enter CPU utilization threshold (default 10%): 0.5

Enter CSV file paths (comma separated), or press Enter to use all in 'pidstat\_data/': /home/ubuntu/pidstat\_data/pid\_22843\_info.csv,/home/ubuntu/pidstat\_data/pid\_29475\_info.csv

Successfully created the PDF '/home/ubuntu/plots-final-check/merged\_output.pdf'