1. Define metrics and thresholds:

It defines a list of metrics to monitor, such as CPU usage, memory usage, disk usage, uptime, and services running.

Each metric also has a corresponding threshold, specifying a limit beyond which it's considered critical.

2. Capture and analyze metrics:

The script retrieves data for each metric using various commands specific to the chosen metric (e.g., top for CPU, free for memory).

It then compares the captured value to the defined threshold.

3. Report and store results:

If a metric exceeds its threshold or certain services are not running, the script logs the metric and its value to the "metrics.txt" file.

This file serves as a record of the monitored data and can be analyzed later to identify potential performance issues or service outages.

4. Continuous monitoring:

The script runs continuously in a loop, periodically refreshing and capturing new data for all metrics.

This allows for real-time monitoring and continuous updates of the "metrics.txt" file.

Overall, the script's intent was to provide a tool for monitoring my system's performance and health, identifying potential issues through exceeding thresholds or service disruptions.