

**check\_disk\_usage:**

The function checks the disk usage by retrieving the disk usage percentage for the root filesystem, compares it to a predefined threshold, and logs an alert message if the threshold is exceeded.

It uses 'df' to display information about the disk space usage on the root filesystem ("/"). The '-h' flag makes the output human-readable, showing sizes in a more user-friendly format (e.g., "10G" for 10 gigabytes). It uses 'awk' to select row: 2, col: 5 output and sed to remove the '%' symbol from it.

**check\_cpu\_usage:**

This function uses the top command to retrieve the current CPU usage percentage, compares it to a predefined threshold, and logs an alert message if the threshold is exceeded.

It uses the 'top -b' command to get CPU usage (in batch) snapshots, and '-n 1' takes one snapshot. The 'grep' filters the lines containing the '%Cpu(s)' text and then awk retrieves the text and 'cut -d.' manipulates text by cutting from '.' and '-f1' gives the first half of cut word.

**check\_memory\_usage:**

It retrieves the available memory in megabytes, compares it to a predefined threshold, and logs an alert message if the available memory falls below the threshold.

The 'free' command displays the information about memory usage in megabytes and then 'awk' retrieves the relevant field by searching for the line containing Mem format and then calculates memory usage by dividing the 3 col entry by 2 col entry and multiplying by 100.

**rotate\_log\_file:**

function checks the size of a specified log file and, if it exceeds a predefined threshold, rotates the log file by renaming it with a timestamp and creating a new empty log file. This is a common technique used to

prevent log files from growing too large and consuming excessive disk space while retaining a historical record of log data.

In the if condition it checks the existence of the file by the '-f' command and 'du -m "\$log\_file\_path' checks the disk/space used by the specified file and cuts to get the desired part of information. Then it checks it with the threshold size of the log file. If the size is greater then use the 'mv' command to rename the existing log file by appending a timestamp (in the format YYYYMMDDHHMMSS) and the ".bak" extension to it and create a new log file. This effectively creates a backup of the old log file with a timestamp.