Server Metrics Monitoring Shell Script

Bash script creates a menu-driven system for displaying various system information on a Linux-based operating system. When executed, the script presents a menu with different options, and the user can select a specific option to view the corresponding system information

```
#!/bin/bash

# function to show memory usage

memoryUsage(){

    echo "Memory Usage:"

    sudo free -h

    read -p "Press any key to Continue...."

}


# function to show CPU usage

cpuUsage(){

    echo "CPU Usage:"

    sudo top -b -n1 | grep "Cpu(s)"

    read -p "Press any key to Continue...."

}


# function to show user name
```

```
userName(){
	echo "user name:"
	sudo whoami
	read -p "Press any key to Continue...."
}

# function to show hostname
hostName(){
	echo "Hostname:"
	sudo hostname
	read -p "Press any key to Continue...."
}

# function to show virtual memory statistic
virtualmemoryStatistics(){
	echo "virtual memory statistic:"
	sudo vmstat
	read -p "Press any key to Continue...."
}

# function to show PWD
pwd(){
```

```
        echo "PWD:"
        sudo pwd
        read -p "Press any key to Continue...."
}

# function to show total disk utilization
diskUtilization(){
        echo "Total disk utilization:"
        sudo df -h --total|grep total
        read -p "Press any key to Continue...."
}

# function to list block devices
listblockDevices(){
        echo "list block devices:"
        sudo lsblk
        read -p "Press any key to Continue...."
}

# function to show list all tcp ports
listtcpPorts(){
        echo "list all tcp ports:"
```

```
        sudo netstat -at
        read -p "Press any key to Continue...."
}

# function to show list all udp ports
listudpPorts(){
        echo "list all udp ports:"
        sudo netstat -au
        read -p "Press any key to Continue...."
}

# function to show kernel version
kernelVersion(){
        echo "show kernel version:"
        sudo uname -r
        read -p "Press any key to Continue...."
}


# function to show os name & version
osVersion(){
        echo "show OS name and version:"
```

```
    sudo cat /etc/os-release |grep VERSION= && cat /etc/os-release |grep NAME
    read -p "Press any key to Continue...."
}

# function to show server uptime
serverUptime(){
    echo "show server uptime:"
    sudo uptime -p
    read -p "Press any key to Continue...."
}


# function to View Partition on a /dev/xvda Disk
diskPartition(){
    echo "View Partition on a /dev/xvda Disk:"
    sudo fdisk -l /dev/xvda
    read -p "Press any key to Continue...."
}

# function to show menu
show_menu()
{
```

```
        clear
        echo "++++++++++++ MENU ++++++++++++"
        echo "1. memory usage"
        echo "2. cpu usage"
        echo "3. username"
        echo "4. hostname"
        echo "5. virtual memory statistic"
        echo "6. pwd"
        echo "7. total disk utilization"
        echo "8. list block devices"
        echo "9. list all tcp ports"
        echo "10. list all udp ports"
        echo "11. kernel version"
        echo "12. os name & version"
        echo "13. server uptime"
        echo "14. View Partition on a /dev/xvda Disk"
        echo "15. Enter q to quit"
        echo "++++++++++++++++++++++++++++++"
}

# function to take input
take_input()
```

```
{
    #take the input and store it in choice variable
    local choice
    read -p "Select the option from above menu: " choice

    #using switch case statement check the choice and call function.
    case $choice in
        1) memoryUsage ;;
        2) cpuUsage ;;
        3) userName ;;
        4) hostName ;;
        5) virtualmemoryStatistics ;;
        6) pwd ;;
        7) diskUtilization ;;
        8) listblockDevices ;;
        9) listtcpPorts ;;
        10) listudpPorts ;;
        11) kernelVersion ;;
        12) osVersion ;;
        13) serverUptime ;;
        14) diskPartition ;;
        q) exit 0;;
```

```
        *) echo "Enter Valid Option!!"
            read -p "Press any key to Continue...."

        esac
    }

# for loop to call the show_menu and take_input function.
while true
do
    show_menu
    take_input
done
```

Explanation:

The script presents a menu with different options, and the user can select a specific option to view the corresponding system information.

1.The script defines multiple functions, each responsible for displaying a specific type of system information. These functions include memory usage, CPU usage, user name, hostname, virtual memory statistics, PWD (present working directory), disk utilization, list of block devices, list of TCP ports, list of UDP ports, kernel version, OS name and version, server uptime, and disk partition information.

2.The show_menu function clears the screen and displays a menu with numbered options representing each type of system information.

3.The take_input function prompts the user to select an option from the menu. It reads the user's input and uses a switch case statement to call the corresponding function based on the selected option.

4.The script enters an infinite loop using a while true loop. Inside the loop, it calls the show_menu function to display the menu and then calls the take_input function to handle user input and execute the chosen function.

5.Each function displays the relevant system information by executing specific commands with elevated privileges using sudo.

6.After displaying the information, the script pauses and waits for the user to press any key before returning to the menu.

By running this script, you can interactively choose which system information you want to view by selecting the corresponding option from the menu. You can keep selecting different options until you choose to exit the script by selecting option "15" or entering "q".

Please note that running commands with elevated privileges (sudo) may prompt for a password, depending on your system configuration. Make sure you understand and trust the commands being executed before running the script.