

# **System Monitoring Report with my Practical Findings**

## **System Overview:**

The system under consideration is a remote server running Ubuntu Server 20.04 LTS. It serves as a versatile platform for my personal computing needs. As an avid user of Ubuntu on my personal workstation, I take great interest in monitoring and optimizing system performance. In this report, I will provide insights into my personal system monitoring practices along with command usage to demonstrate my hands-on experience with system management.

### **1. Monitoring System Resources:**

- **top Command Output:**
  - The `top` command provided me real-time insights into my system resource utilization. It displayed information such as CPU load, memory usage, and a list of running processes. The display is continuously updated and it allows us for continuous monitoring of our system performance.
- **free -m Command Output:**
  - Upon executing the `free -m` command, I obtained memory usage statistics in megabytes. This included details on total memory, used memory, and free memory, providing a comprehensive view of system memory utilization.
- **df -h Command Output:**
  - The `df -h` command yielded information on disk space usage in a human-readable format. It presented data on total disk space, used space, and available space on each filesystem, enabling easy identification of disk usage patterns.
- Running the `iostat` command provided me with information about how data was being read from and written to the disk. This helped me find any issues that might slow down disk performance by showing which programs were using a lot of disk resources.

### **2. Checking System Uptime:**

- I frequently checked how long my system had been running smoothly without any unexpected interruptions. I did this by using the `uptime` command. It showed me the time since the system was last booted up, which helped me ensure that everything was working fine without any issues..

### 3. Viewing Log Files:

- Since yesterday, I've been using log files to troubleshoot and monitor system events. Whenever I need to keep an eye on what's happening on my system, I turn to these files. For example, I use the `tail /var/log/syslog` command to quickly see the last few lines of the system log file, giving me insight into recent system activities or errors. Additionally, I check the `tail /var/log/auth.log` file to specifically monitor authentication-related messages and user login attempts, ensuring the security of my system and identifying any unauthorized access attempts.

### 4. Network Monitoring:

- In the past, I understood the importance of monitoring network activity, but Since yesterday, I've been actively monitoring my network using various commands:
- `ifconfig`: This command has allowed me to inspect the configuration of my network interfaces and check the assigned IP addresses. It's been helpful in understanding how my system connects to the network.
- `netstat -tuln`: By using this command, I've been able to observe all the active network connections and the ports my system has been listening to. It has provided insights into which services are communicating over the network.
- `ping <hostname>`: I've frequently used this command to test the connectivity to specific hosts on the network. It has helped me ensure that my system can communicate with other devices without any issues.

These actions have been crucial in ensuring the smooth functioning of my network and identifying any potential connectivity issues early on.

### Challenges:

I encounter several challenges like in the following:

**Syntax Errors:** Command syntax can be complex, and a small mistake can lead to errors. Ensuring accurate syntax can be challenging, especially when dealing with commands that have multiple options and arguments.

**Understanding Output:** The output of some commands can be verbose and cryptic, making it difficult to interpret. Understanding and extracting relevant information from command output can be challenging, especially for beginners.

**Permission Issues:** Some commands require administrative privileges to execute, and you may encounter permission errors if you do not have the

necessary permissions. Managing permissions and elevating privileges when needed can be a challenge, especially in multi-user environments.

**Command Options:** Many commands have numerous options and arguments, and remembering all of them can be challenging. Consulting documentation or using the `--help` option can help, but it can still be overwhelming to remember all the available options.

**Risk of Data Loss:** Certain commands, such as `rm` for removing files, can permanently delete data if used incorrectly. Care must be taken when executing such commands to avoid accidental data loss.

**Network Connectivity Issues:** While monitoring network activity, you may encounter issues such as network outages, slow connections, or misconfigured network settings. Troubleshooting these issues can be challenging and may require knowledge of networking concepts.

**System Performance Impact:** Some monitoring commands, especially those that collect real-time data, can impact system performance. Monitoring resource usage and ensuring that monitoring tools themselves do not consume excessive resources can be challenging.

Overall, while these commands are powerful tools for managing a Linux system, they also come with their own set of challenges that require careful attention, motivation and understanding to overcome.

## Conclusion:

In the past, I began actively monitoring my system's health and performance, despite having only a basic understanding of Linux. I started this practice yesterday. This involved regularly checking system resources, uptime, log files, process management, and network activity using Ubuntu's command-line tools.

By gradually building upon my Linux skills and diligently monitoring these areas, I aimed to ensure the stability, efficiency, and security of my personal workstation. This proactive approach allowed me to detect and address any issues promptly, minimizing disruptions and maximizing productivity.

Additionally, actively optimizing system performance helped me maintain a seamless computing experience tailored to my specific needs, even with my limited Linux expertise. With each task performed and each challenge faced, I gained valuable experience and confidence in managing my system effectively.

