MONITORING SYSTEM PROCESSES AND OPTIMIZING PERFORMANCE

Introduction

Efficient process management is crucial for maintaining system performance and stability. Operating systems constantly manage multiple background processes and user-initiated applications, which consume system resources such as CPU, memory, disk, and network bandwidth. By monitoring these processes, users can identify resource-intensive tasks and optimize performance by terminating non-essential applications. This report presents an analysis of my system's running processes, identifies memory-intensive applications, and evaluates which processes are safe to terminate for performance optimization.

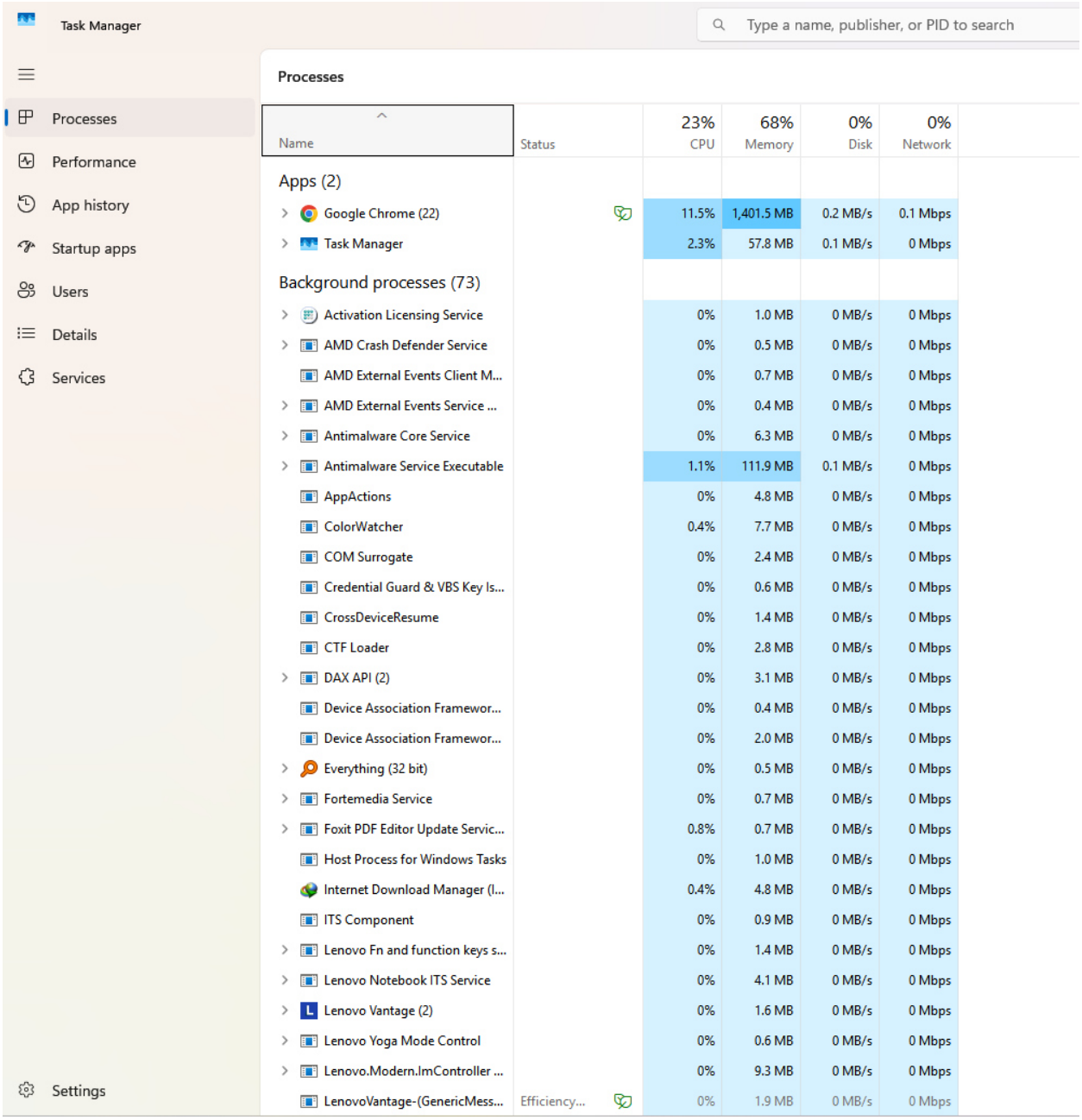
Process Discovery and Analysis

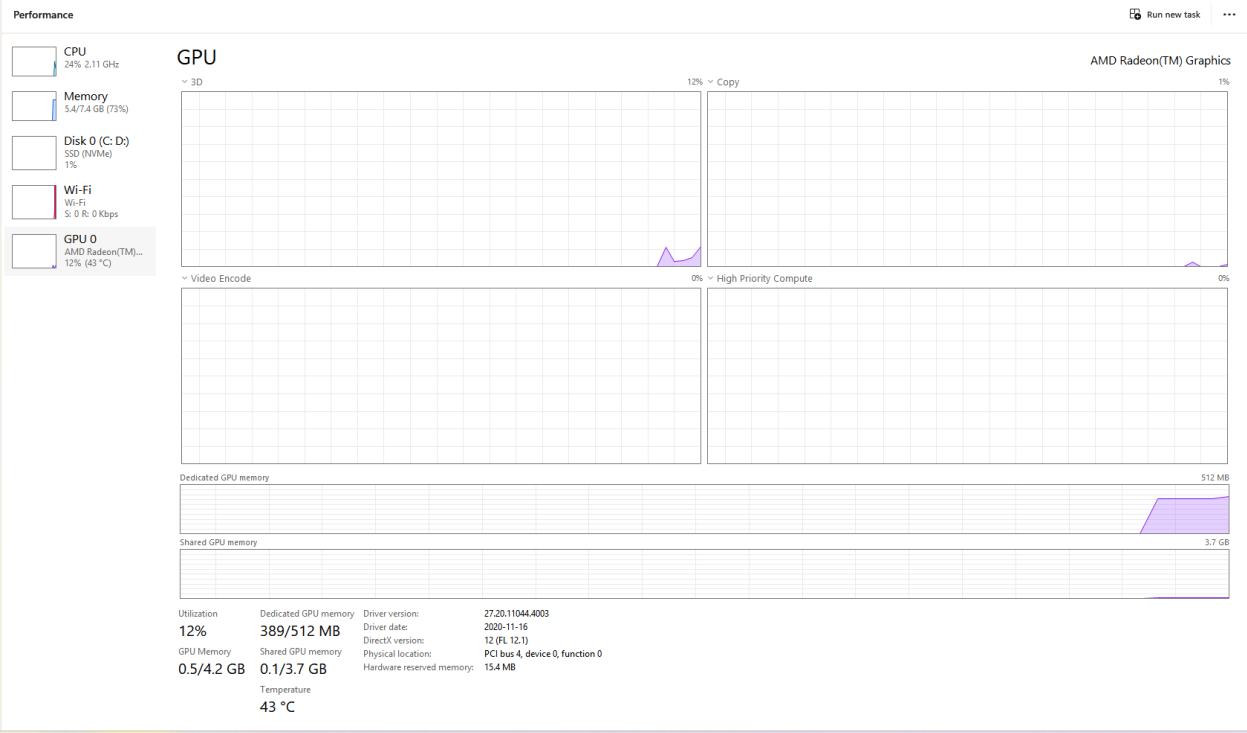
Using Windows Task Manager, I examined the active processes and categorized them into applications and background processes. The Task Manager provides real-time monitoring of CPU, memory, disk, and network usage, enabling users to assess the impact of each process on system resources.

The image attached shows the Task Manager under the “Processes” tab. The system exhibits the following usage levels:

- **CPU Usage**: 23%
- **Memory Usage**: 68%
- **Disk Usage**: 0%

- **Network Usage**: 0%





Among all processes, **Google Chrome** is consuming the most memory, using approximately **1,401.5 MB**. This is not surprising as Chrome is known for its high memory usage due to its multi-process architecture, where each tab and extension runs as a separate process to ensure stability and security (Google, 2023).

The second notable process is **Antimalware Service Executable**, utilizing **111.9 MB** of memory. This service is part of Windows Security and runs in the background to provide real-time protection against malware threats. Other active processes like **Foxit PDF Editor Update Service (0.7 MB)**, **Internet Download Manager (4.8 MB)**, and **Lenovo Vantage (1.6 MB)** are using significantly less memory and pose minimal impact on overall system performance.

Identifying Non-Essential Processes

Determining which processes are safe to terminate requires careful consideration. System-critical processes, such as **Antimalware Service Executable**, **Device Association Framework**, and **COM Surrogate**, should never be turned off as they are integral to system stability and security.

Conversely, third-party applications that are not in active use can be safely closed. For instance, **Foxit PDF Editor Update Service** is a background service that periodically checks for updates but does not need to run continuously. Similarly, **Internet Download Manager (IDM)**, which is primarily used for managing downloads, does not need to remain active when no downloads are in progress.

After reviewing the processes, I decided to terminate **Everything (32-bit)**. This application is a search indexing tool that enhances file search capabilities but is not vital for system operation. Disabling it temporarily would free up resources without compromising system functionality.

Observed Performance Improvement

After terminating the **Everything** process, there was a slight decrease in memory usage. While the overall impact was modest due to the application's small footprint (0.5 MB), the exercise demonstrates the importance of regularly auditing background processes to maintain optimal system performance.

Moreover, closing unused Chrome tabs or extensions could result in a substantial reduction in memory consumption. According to Microsoft's Process Monitor documentation, continuously

running unnecessary background applications can accumulate over time and degrade system responsiveness (Russinovich, 2023).

Conclusion

Through systematic monitoring using Task Manager, I identified Google Chrome and Antimalware Service Executable as the top memory-consuming processes. While essential system processes must remain active, non-critical applications like **Everything** can be safely terminated to free up memory resources. Although the performance gain from terminating a single low-impact process is minor, this practice, when applied regularly, contributes to a more responsive and efficient computing experience. Awareness of resource usage and prudent process management are key to maintaining system health.

References

Google. (2023). *Chrome Multi-Process Architecture*. Google Chrome Developers. https://developer.chrome.com/blog/inside-browser-part1

Russinovich, M. (2023). *Process Monitor v3.92*. Microsoft Sysinternals. https://learn.microsoft.com/en-us/sysinternals/downloads/procmon

Word Count: 573