Linux Performance Analysis Report

- Time: 2025-06-10 06:25:12.146495
- Author: deepseek-ai/DeepSeek-R1

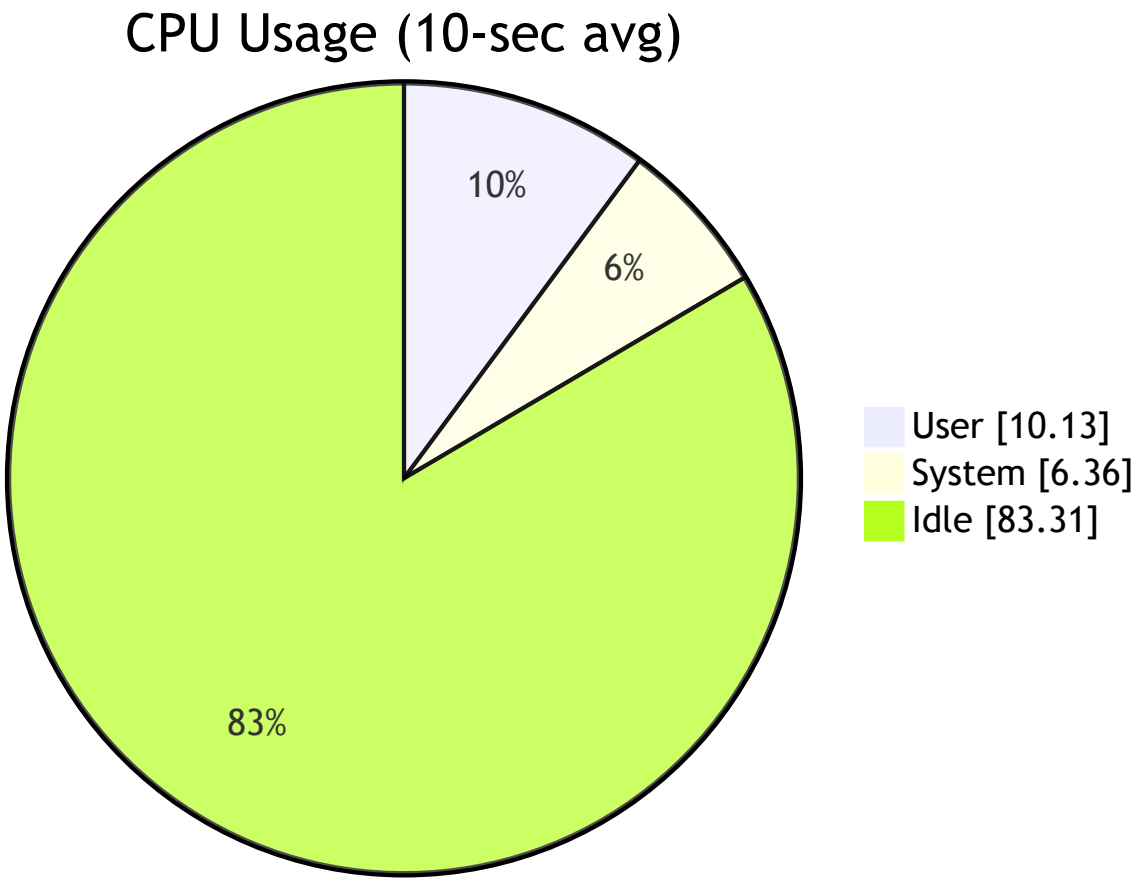
System Performance Analysis Report

System Overview

▶ Host Information

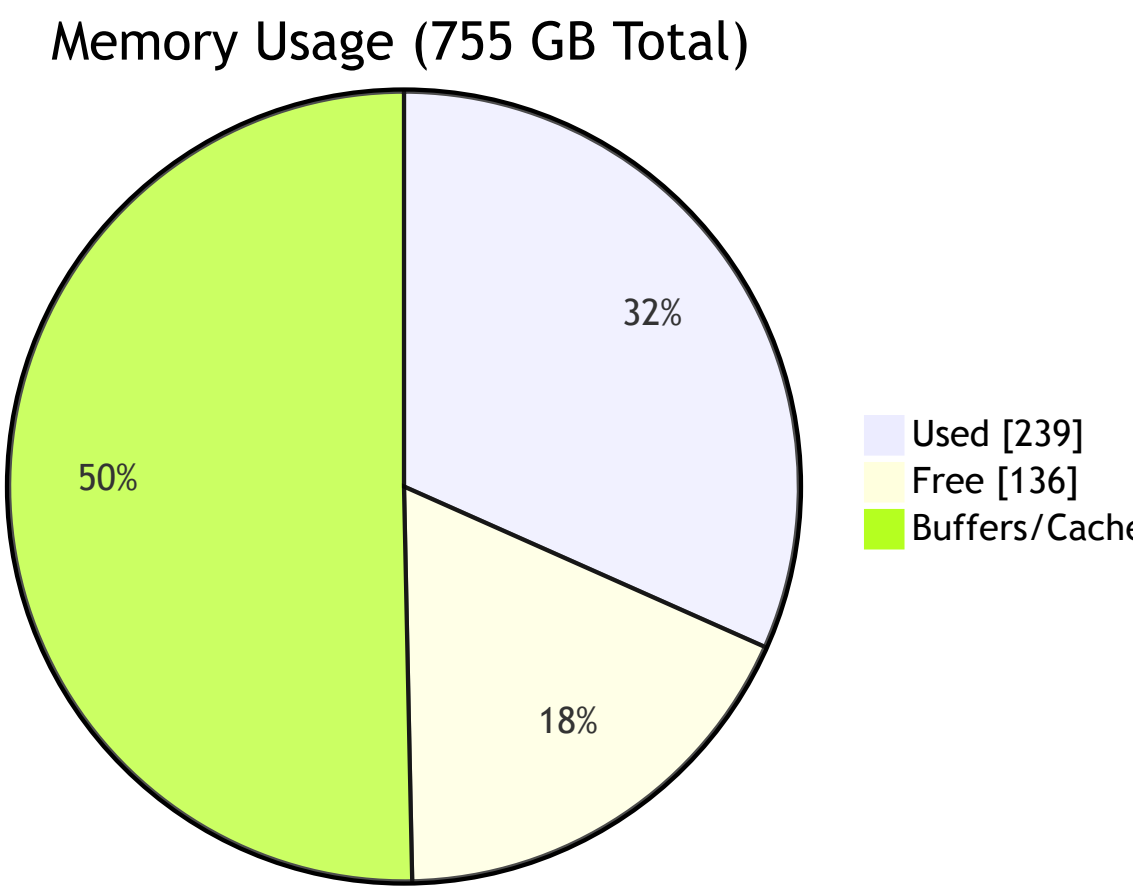
Resource Utilization

CPU Usage



- **Load Average (1-min)**: 20.94
- **Thread Utilization**: 128 threads, 20.94 load → **No CPU saturation** (load < thread count)

Memory Utilization



- **RAM Usage**: 244 GB/755 GB (32.3% usage)
- **Swap Usage**: 0 MB → **No memory pressure**

Disk I/O Performance

Device	IOPS	Throughput	Queue	Util%
nvme2n1	67	852 KB/s	0.01	0.4%
Other	0-6	0-24 KB/s	0.00	0%

- **Max Utilization**: 0.4% → **No disk bottleneck**

Top Resource-Consuming Processes

PID	User	CPU%	MEM%	Command
9850	solana	2070	29	agave-validator
32480	solana	400	0	ps
18871	solana	49	0	jito-shredstrea
802	root	3	0	kcompactd0
16496	solana	0.5	0	node_exporter

Key observations:

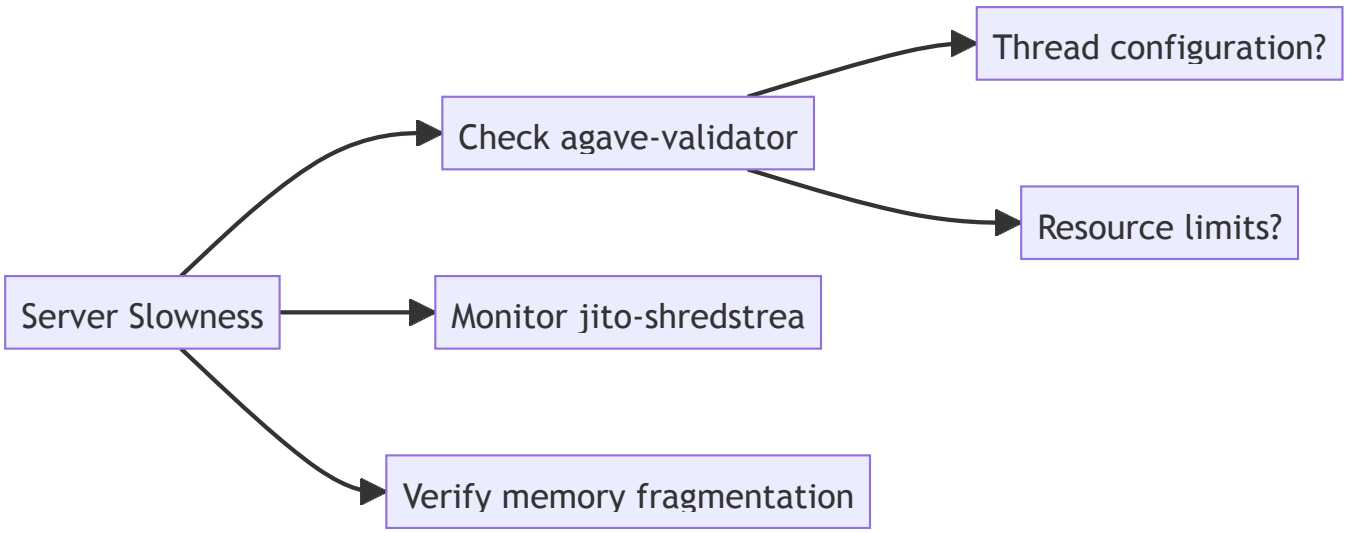
1. `agave-validator` dominates resources (20 CPU cores, 29% RAM)
2. `jito-shredstrea` shows moderate CPU usage
3. Kernel process `kcompactd0` active (memory management)

Analysis & Recommendations

Identified Issues

1. **Extreme Process Imbalance**:
   - Single process (`agave-validator`) using 20 entire CPU cores
   - Indicates inefficient threading or unoptimized workload distribution
2. **Mysterious High CPU Processes**:
   - `ps` process consumed 400% CPU during monitoring → Likely a transient command execution
   - Needs verification if recurring
3. **Memory Fragmentation Hints**:
   - `kcompactd0` activity suggests memory fragmentation
   - Though no swap usage, warrants monitoring

Recommendations



Action Steps:

1. 🔍 **Investigate `agave-validator` :**
   - Check thread configuration with: `pstree -p 9850 | wc -l`
   - Review resource limits: `cat /proc/9850/limits`
   - Consider CPU affinity tuning via `taskset`
2. 🧩 **Monitor Memory Fragmentation:**
   bash
   cat /proc/buddyinfo
   cat /proc/pagetypeinfo
3. ⚠️ **Identify `ps` Command Origin:**
   - Check audit logs: `grep 32480 /var/log/auth.log`
   - Verify cron jobs: `grep -r solana /etc/cron*`
4. 🔄 **Optimize Jito Processes:**
   - Profile I/O patterns: `strace -p 18871 -f -e trace=file`
   - Monitor network usage if blockchain-related

> **Critical Note**: No hardware-level bottlenecks found. Issue likely in application-layer workload distribution - focus on `agave-validator` optimization.