Log Collector Hardware Requirements Guide

What is a Log Collector?

A log collector is a tool or software component designed to gather log data from various sources within an IT environment, including servers, applications, network devices, and other infrastructure components. The primary purpose is to centralize log data for analysis, monitoring, and troubleshooting.

Key Considerations

- **Always Online**: The log collector should be online at all times to ensure continuous collection of logs from various sources.
- **Dedicated Unit**: It's best to use a separate or dedicated unit for the log collector to avoid interference with other systems.
- **Virtual Machine (VM)**: Preferably, the log collector should be set up as a virtual machine for flexibility and ease of management.
- **High Availability**: Consider implementing redundancy to prevent log collection disruption during maintenance or failures.
- **Geographical Distribution**: For global organizations, consider deploying regional log collectors to minimize network latency and bandwidth usage.

Hardware Requirements

When setting up a log collector (such as Logstash) to handle multiple log sources, consider the following hardware specifications:

CPU

- **Minimum**: 4 CPU cores
- **Optimal**: 4-8 CPU cores with 2GHz+ on each core
- **Enterprise-level**: 8-16 cores for high-volume environments (10,000+ events per second)
- **Note**: Logstash is CPU-intensive, especially when processing complex pipelines with multiple filters

- **Scaling factor**: Add approximately 1-2 cores for every additional 5,000 events per second

Memory (RAM)

- **Minimum**: 8 GB RAM
- **Optimal**: 16 GB RAM or more
- **Enterprise-level**: 32-64 GB for high-volume environments
- **Note**: Additional memory may be required when processing large volumes of data or using memory-intensive filters
- **JVM considerations**: If using Java-based collectors, allocate 50-70% of system memory to the JVM heap

Storage

- **Minimum**: 100 GB disk space
- **Optimal**: 500 GB to 1 TB of disk space
- **Enterprise-level**: 2-4 TB with RAID configuration for high availability
- **Recommendation**: Fast disks (SSD) for better performance, especially if using persistent queues
- **IOPS requirements**: At least 3,000 IOPS for high-volume environments
- **Temp storage**: Additional 20-30% space for temporary file storage and buffer overflow protection
- **Note**: Storage requirements depend on log volume and retention policies

Network

- **Requirement**: One or more reliable network adapters
- **Bandwidth**: At least 1 Gbps for medium-sized environments
- **Enterprise-level**: 10 Gbps networking for high-volume environments
- **Redundancy**: Dual NICs configured for failover
- **Note**: Ensure your network can handle the data throughput from all log sources
- **Network isolation**: Consider a dedicated VLAN for log collection traffic

Operating System

- **Compatible with**: Linux distributions such as Red Hat Enterprise Linux (RHEL), CentOS, or Ubuntu
- **Windows support**: Windows Server 2016 or later if using Windows-based collectors
- **Virtualization**: VMware ESXi, Hyper-V, or KVM for virtualized environments
- **Note**: Ensure your OS is up-to-date and compatible with your log collector software

- **Kernel parameters**: Adjust file descriptor limits and network buffer sizes for optimal performance

Additional Software Requirements

- **Java**: If using Logstash, it runs on the Java Virtual Machine (JVM). Recent Logstash versions include a bundled JDK.
- **Database**: Some log collectors require a database backend (PostgreSQL, MongoDB) for metadata storage
- **Container support**: Docker or Kubernetes for containerized deployments
- **Monitoring tools**: Prometheus, Grafana, or similar for monitoring collector performance

Performance Considerations

- **Log volume**: Calculate expected events per second (EPS) and size per event
- **Parsing complexity**: Complex regex and transformation operations require more CPU
- **Queue sizing**: Memory queues vs. persistent queues (disk-based) affect performance and durability
- **Batching**: Adjust batch sizes for optimal throughput (typically 125-1000 events per batch)
- **Pipeline workers**: Configure parallel processing based on available CPU cores
- **Compression**: Enable compression for network transfer to reduce bandwidth requirements
- **Buffer sizing**: Configure adequate buffer sizes to handle traffic spikes

Benefits of Proper Hardware Configuration

- **Centralized Logging**: A single log collector simplifies monitoring and analyzing logs from different sources.
- **Improved Security**: Continuous log collection helps in identifying and responding to security incidents promptly.
- **Enhanced Performance**: Using a dedicated unit or VM ensures that the log collector operates efficiently without affecting other systems.
- **Regulatory Compliance**: Proper log collection infrastructure helps meet compliance requirements (GDPR, HIPAA, PCI DSS).
- **Operational Intelligence**: Enables better decision-making through comprehensive visibility into system operations.

Additional Considerations

- **Load Testing**: Before finalizing your hardware setup, conduct load testing to simulate the expected log volume and identify potential bottlenecks.

- **Scalability**: Plan for growth by choosing hardware that can be easily upgraded or by deploying log collectors in a distributed setup.
- **Capacity Planning**: Forecast log growth over time and plan for hardware upgrades accordingly.
- **Backup Strategy**: Implement regular backups of log collector configuration and critical data.
- **Disaster Recovery**: Plan for quick recovery in case of collector failure.
- **Security Hardening**: Apply security best practices to protect the log collector itself.
- **Monitoring**: Implement monitoring of the log collector's health and performance.
- **Alerting**: Set up alerts for collector-related issues like queue saturation or processing delays.

Architecture Patterns

Tiered Collection

- **Edge collectors**: Lightweight collectors at source locations
- **Aggregation layer**: Midtier collectors that receive data from edge collectors
- **Central storage**: Final destination for processed logs

Load Balancing

- **Distributed intake**: Multiple intake nodes behind a load balancer
- **Shared processing**: Distribute processing load across multiple worker nodes
- **Clustered storage**: Distributed storage backend for log data

Specialized Processing

- **Pre-processors**: Dedicated nodes for initial parsing and filtering
- **Enrichment nodes**: Add context and metadata to logs
- **Analytics nodes**: Specialized hardware for complex analysis operations

If you need further assistance, kindly contact our support at ***support@cytechint.com*** *for prompt assistance and guidance.*

Revision #3
Created 24 April 2025 12:59:50 by Richmond Abella
Updated 28 May 2025 09:04:23 by Richmond Abella