**1. Server crash due to high CPU usage**

**Action**:

* Investigate running processes consuming excessive CPU.
* Optimize processes and monitor the system regularly.

**Recommendation**:

* Identify resource-hogging processes and either optimize them or distribute the load across additional servers.
* Implement alerting thresholds for CPU usage to catch high usage early.

**2. Memory leak causing slow performance**

**Action**:

* Restart the application or server to clear memory.
* Investigate the code for potential memory leaks.

**Recommendation**:

* Perform a detailed code audit to identify and resolve memory leaks.
* Use profiling tools to monitor memory usage and ensure no resources are held unnecessarily.

**3. Disk full due to excessive logs**

**Action**:

* Clean up old logs or move them to external storage.
* Implement log rotation and monitoring.

**Recommendation**:

* Set up log rotation policies to automatically archive or delete logs after a certain period.
* Consider using a centralized log management system to handle logs more efficiently.

**4. Application timeout due to high CPU usage**

**Action**:

* Optimize API endpoints causing high CPU load.
* Scale server resources to handle peak loads.

**Recommendation**:

* Investigate API bottlenecks and refactor or optimize resource-intensive endpoints.
* Use load balancing to distribute traffic evenly across servers to avoid overloading any single server.

**5. Unexpected server shutdown due to power failure**

**Action**:

* Investigate power supply issues.
* Install a UPS (Uninterruptible Power Supply) for backup power.
* Monitor for future occurrences.

**Recommendation**:

* Implement redundancy in power systems to prevent future outages.
* Set up a monitoring system to alert if power fluctuations are detected.

**6. High memory usage causing system slowdowns**

**Action**:

* Check for memory-hogging applications.
* If necessary, increase memory allocation or optimize memory-heavy processes.

**Recommendation**:

* Implement memory management best practices such as limiting the number of concurrent applications or processes.
* Consider adding more physical memory to the server if the system routinely runs out of memory.

**7. Network connectivity issues**

**Action**:

* Check network cables, switches, and routers.
* Analyze network traffic for bottlenecks.
* Consider upgrading network infrastructure.

**Recommendation**:

* Perform regular network health checks and replace any failing hardware.
* Monitor network performance to identify congestion