**Case#01 Title: Enhancing Availability and Scalability for ShopNext E-Commerce Platform**

**Background:**

ShopNext is a mid-sized e-commerce company experiencing rapid growth. With increased website traffic during seasonal sales, the company started facing issues such as server overload, slow response times, and occasional outages. The company decided to overhaul its infrastructure to ensure high availability (HA) and better scalability.

**Problem Statement:**

The legacy infrastructure was hosted on a single powerful server. Although it had a strong CPU and RAM (vertical scaling), it couldn’t efficiently handle spikes in traffic. There was no redundancy or backup in case the server went down.

**Goals:**

* Achieve near-zero downtime during peak usage
* Improve user experience with faster page loading
* Scale the system to meet demand without manual intervention

**Solution Implemented:**

**1. Load Balancing:**

A load balancer (NGINX) was deployed to distribute incoming traffic across multiple web servers. This eliminated single-point failure and improved system responsiveness.

**2. Horizontal Scaling:**

Instead of adding more power to one server, additional web and application servers were added to the pool. These servers were configured identically and placed behind the load balancer.

**3. Auto-Scaling:**

Using Amazon Web Services (AWS), ShopNext implemented auto-scaling groups. Servers were automatically added or removed based on metrics like CPU usage and response time.

**4. Database Clustering:**

The database was shifted to a managed SQL cluster that supported read replicas, improving performance and redundancy.

**Outcome:**

* Page load times improved by 40%
* The system maintained 99.98% uptime during Black Friday sales
* Infrastructure costs were optimized using on-demand scaling
* Customer satisfaction and sales conversion rates increased

**Discussion Questions for Students:**

1. What are the key differences between horizontal and vertical scaling in this case?
2. How does a load balancer contribute to fault tolerance?
3. What risks would ShopNext face without implementing auto-scaling?
4. Suggest additional ways the company could further improve reliability.
5. How does high availability benefit the end users of an application?

**Case #02 Title:** **Enhancing Stability and Security of a University Management System through Software Maintenance**

**Background:** UIT University's IT department developed a custom University Management System (UMS) in 2020 for handling student data, fee collection, course management, and faculty records. Initially, the system was stable, but within 18 months, performance issues, occasional crashes, and multiple security warnings started surfacing. This case study explores how applying proper maintenance strategies helped restore the system’s reliability and security.

**Problem Statement:** The UMS began showing vulnerabilities due to outdated libraries, missing patches, and increased user demand. Students experienced login errors, delayed result processing, and data inconsistencies.

**Applied Maintenance Strategies:**

1. **Corrective Maintenance:**
   * Immediate resolution of bugs that were causing result upload failures.
   * Error logs were reviewed weekly to identify recurring faults.
2. **Adaptive Maintenance:**
   * Integrated new payment gateway after discontinuation of the older one.
   * Updated compatibility with the latest version of Chrome and Firefox.
3. **Perfective Maintenance:**
   * Refactored some modules to reduce memory usage.
   * Improved user interface based on student feedback.
4. **Preventive Maintenance:**
   * Set up automated scripts for weekly vulnerability scanning.
   * Introduced regular dependency updates and codebase documentation.
5. **Automated Patching:**
   * Used tools like WSUS and Patch My PC for OS-level and third-party updates.
   * Created a patch testing environment before live deployment.

**Outcome:**

* Downtime was reduced by 75%.
* Reported bugs declined significantly after corrective updates.
* Security score improved due to proactive patching.
* The UMS is now used by over 3,000 students with minimal complaints.

**Discussion Questions:**

1. What would have happened if the preventive maintenance strategy was ignored?
2. How does automated patching reduce human error?
3. Which maintenance type do you think was most crucial in this case? Why?
4. Suggest additional tools that could support ongoing maintenance.
5. How can user feedback play a role in perfective maintenance?

**Case #03 Title:: Incident Response and Root Cause Analysis**-Server Outage Due to Unauthorized Access – An Incident Response Exercise

**Background:** XYZ Tech Ltd. is a mid-sized software development company with cloud-hosted infrastructure. One Monday morning, developers found that their main project management system was down and returning 403 Forbidden errors to all users.

**Incident Response Timeline:**

**09:00 AM** – Users report system inaccessibility via the internal helpdesk.

**09:15 AM** – IT team starts initial investigation; confirms services are down across multiple departments.

**09:30 AM** – Incident Response Team (IRT) is activated.

**09:45 AM** – Logs show unauthorized login attempts followed by a successful login to the admin panel.

**10:00 AM** – Affected servers are isolated; suspicious IP address is blocked.

**11:00 AM** – Backup services are triggered to restore system state from the previous night.

**02:00 PM** – Services are restored, but monitoring continues for unusual behavior.

**Root Cause Analysis (RCA):**

* The attacker used leaked employee credentials (phishing incident reported last week).
* No multi-factor authentication (MFA) was enabled on the admin panel.
* Log monitoring alerts were turned off due to a recent reconfiguration.

**Post-Mortem Analysis:**

**Findings:**

* Lack of MFA and poor password hygiene contributed to the breach.
* IRT responded within 15 minutes, reducing potential damage.

**Recommendations:**

* Enforce MFA on all privileged accounts.
* Conduct periodic phishing simulations and training.
* Implement SIEM for real-time log alerts and anomaly detection.
* Review and update incident response plan quarterly.

**Discussion Questions for Students:**

1. Was the response time appropriate? Justify your reasoning.
2. What key tools and techniques would improve their response next time?
3. How would you draft an improved incident response plan for XYZ Tech Ltd.?
4. Why is documentation and post-mortem reporting essential after an incident?