# System Configuration and Troubleshooting Manual

## Table of Contents

## 1. Introduction

This manual provides comprehensive guidelines for configuring and maintaining the **X-Server System**, with a focus on performance optimization and issue resolution. The following sections include both detailed configuration instructions and troubleshooting techniques.

## 2. System Architecture Overview

The **X-Server System** is designed with a modular architecture that consists of several subsystems, each responsible for distinct functionalities. The architecture is divided into the following core components:

- **Kernel**: Manages system resources, hardware abstraction, and security enforcement.
- **Network Interface**: Handles external communications and network data exchange protocols.
- **Storage Subsystem**: Responsible for file system management and disk I/O operations.
- **Application Layer**: Runs user applications and integrates with system resources.

### 2.1 Key Technologies

- **Kernel Modules**: The kernel is extensible with various modules, such as `networking`, `storage`, and `security`, to tailor the system to specific use cases.
- **Networking Protocols**: Supports TCP/IP, HTTP/HTTPS, and custom socket-based communication protocols.
- **Filesystem Support**: Provides native support for both local and networked storage solutions, including **NFS**, **iSCSI**, and **RAID** configurations.

## 3. Configuration Settings

### 3.1 Network Configuration

To configure the network interface for optimal performance, the following steps must be executed:

1. **Identify Network Interfaces**: Use the `ifconfig` or `ip addr` command to list all active network interfaces on the system. Example: $ ip addr

2. **Set Static IP Address**: To assign a static IP to an interface, modify the `/etc/network/interfaces` file: Example: auto eth0 iface eth0 inet static address 192.168.1.10 netmask 255.255.255.0 gateway 192.168.1.1

3. **Enable and Restart Networking**: Once the configuration file is updated, restart the networking service: Example: $ sudo systemctl restart networking

## 3.2 Security Configurations

Securing the system is essential for preventing unauthorized access. Recommended steps include:

1. **SSH Configuration**: Edit `/etc/ssh/sshd_config` to disable password-based login: Example: PermitRootLogin no PasswordAuthentication no

2. **Firewall Rules**: Use `iptables` or `ufw` to configure basic firewall rules: Example: $ sudo ufw enable $ sudo ufw allow ssh

## 3.3 Storage Configuration

For configuring the storage subsystem, particularly for large databases or high-availability environments, follow the steps below:

1. **Mounting New Disks**: To add a new disk to the system, use the following commands: Example: $ sudo fdisk /dev/sdb $ sudo mkfs.ext4 /dev/sdb1 $ sudo mount /dev/sdb1 /mnt/storage

2. **Automating Mount at Boot**: To ensure the disk is mounted at boot time, add an entry to `/etc/fstab`: Example: /dev/sdb1 /mnt/storage ext4 defaults 0 2

---

# 4. Troubleshooting

## 4.1 Network Connectivity Issues

If the system is unable to connect to the network, follow these steps:

1. **Check Interface Status**: Verify the network interface is up: Example: $ ip link show eth0

2. **Check Routing Table**: Ensure that the correct route exists: Example: $ ip route show

3. **Ping Gateway**: Test connectivity to the gateway: Example: $ ping 192.168.1.1

## 4.2 Memory Usage Optimization

To reduce memory consumption, review the running processes and adjust their memory limits:

1. **Check Memory Usage**: Use the `free` command to monitor memory usage: Example: $ free -h

2. **Optimize Application Performance**: Modify the configuration of memory-intensive applications to use less memory or swap: Example: $ ulimit -m 2048000

## 4.3 Disk Space Management

To manage disk space, consider the following steps:

1. **Check Disk Usage**: Use the `du` command to identify large files and directories: Example: $ du -h /var/log

2. **Clean Up Log Files**: Remove old log files to free up space: Example: $ sudo rm -rf /var/log/old_log*

---

# 5. Performance Tuning

## 5.1 CPU Performance

Optimizing CPU performance involves adjusting process priorities and managing CPU-bound tasks:

1. **Monitor CPU Usage**: Use the `top` or `htop` commands to identify CPU-heavy processes: Example: $ top

2. **Adjust Process Priority**: Use the `nice` and `renice` commands to change the priority of processes: Example: $ sudo renice -n -10 1234

## 5.2 I/O Performance

For high I/O performance, consider the following adjustments:

1. **Tune Disk Scheduler**: Change the disk I/O scheduler to `noop` for lower latency: Example: $ echo noop > /sys/block/sda/queue/scheduler

2. **Monitor Disk Latency**: Use `iostat` to monitor disk performance: Example: $ iostat -x 1

---

# 6. FAQ

**Q: How do I reset the system to its default configuration?**
A: You can reset the system settings by reverting the configuration files in `/etc/` to their default values. A backup is recommended before making changes.

**Q: What should I do if the system becomes unresponsive?**
A: First, attempt to access the system via a remote terminal. If this fails, reboot the system in single-user mode for further diagnosis.

---

# 7. Glossary

- **Kernel**: The core of an operating system that manages system resources.
- **SSH**: Secure Shell, a protocol for securely accessing remote systems.
- **iptables**: A user-space utility program for configuring Linux kernel firewall.

---