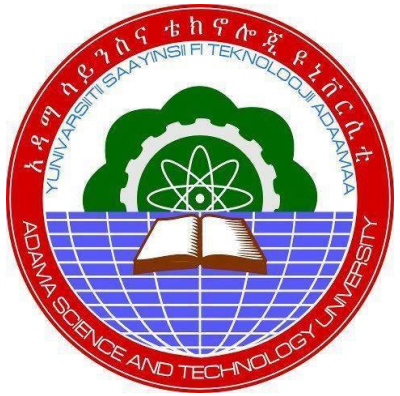


Chapter –6

Software Administration

SEng4203

Lecture notes
ASTU, 2023
Begna D.
ibmbegna@gmail.com

CHAPTER OUTLINES

- System logs, updates, patches, and configuration changes, backups.
- Installing and configuring hardware and software.
- System performance tuning
- Performing routine audits of systems and software.
- Backup

SYSTEM LOGS, UPDATES, PATCHES, AND CONFIGURATION CHANGES, BACKUPS.

- System logs, updates, patches, and configuration changes, and backups, are crucial components of software administration during maintenance and evolution.
- **System logs:** record important **events**, **activities**, and **errors** that occur within a software system.
- They serve as a valuable source of information for troubleshooting issues, identifying errors, and monitoring system behavior.
- Software administrators should regularly review and analyze system logs to gain insights into the system's performance, diagnose problems, and take appropriate actions to address them.

SYSTEM LOGS, UPDATES, PATCHES, AND CONFIGURATION CHANGES, BACKUPS..

- **Updates and patches:** Updates and patches are essential for maintaining the security, stability, and functionality of a software system.
- Updates often include bug fixes, performance improvements, and new features.

SYSTEM LOGS, UPDATES, PATCHES, AND CONFIGURATION CHANGES, BACKUPS..

- **Patches** specifically focus on addressing security vulnerabilities.
- Software administrators should stay updated with the latest updates and patches provided by software vendors or open-source communities and ensure they are promptly applied to the system to mitigate potential risks and keep the software up-to-date.

SYSTEM LOGS, UPDATES, PATCHES, AND CONFIGURATION CHANGES, BACKUPS..

- **Configuration changes:** involve **modifying the settings and parameters** of a software system to adapt it to evolving requirements or optimize its performance.
- Software administrators should carefully manage configuration changes, documenting them appropriately and maintaining a version history.
- It is important to ensure that changes are tested thoroughly, and their impact on the system is assessed before deploying them to production environments.

SYSTEM LOGS, UPDATES, PATCHES, AND CONFIGURATION CHANGES, BACKUPS..

- **Backups:** are essential for data protection and disaster recovery.
- It involve creating copies of critical system data and storing them securely in separate locations.
- Regular backups enable software administrators to restore the system to a previous state in case of data loss, system failures, or security breaches.
- It is important to establish backup schedules, automate the backup process where possible, and periodically validate the integrity and recoverability of the backup data.

SYSTEM LOGS, UPDATES, PATCHES, AND CONFIGURATION CHANGES, BACKUPS..

- Automating the process of system backups can help ensure that they are performed regularly and reliably.
- Here are some steps you can take to automate the backup process:
- **Define backup requirements:** Start by determining what data and system components need to be backed up.
- Identify critical files, databases, configurations, and any other important assets that should be part of the backup process.

SYSTEM LOGS, UPDATES, PATCHES, AND CONFIGURATION CHANGES, BACKUPS..

- **Choose a backup solution:** Select a backup solution that meets your requirements.
- There are various options available, including cloud-based backup services, backup software, or built-in backup utilities provided by the operating system or database management system.
- Consider factors such as scalability, compatibility, ease of use, and security when choosing a solution.
- **Establish a backup schedule**: Define a backup schedule that suits your needs.
- Determine the frequency of backups based on the rate of data changes, importance of the data, and available resources.
- Common backup schedules include daily, weekly, or monthly backups.
- Ensure that the schedule allows sufficient time for backup completion.

SYSTEM LOGS, UPDATES, PATCHES, AND CONFIGURATION CHANGES, BACKUPS..

- **Automate backup execution**: Set up automation scripts or tools to initiate and manage the backup process.
- Depending on the chosen backup solution, there may be built-in automation features or APIs available for integration.
- Use these capabilities to schedule and trigger backups automatically at the defined intervals.
- **Include error handling and notifications:** Implement error handling mechanisms and notifications to ensure the reliability of the backup process.
- Configure the backup automation script to identify and handle any errors or failures that may occur during the backup process.
- Set up notifications to alert you of any backup failures or issues, enabling you to take immediate action.

SYSTEM LOGS, UPDATES, PATCHES, AND CONFIGURATION CHANGES, BACKUPS..

- **Monitor backup status:** Regularly monitor the status of backup operations to ensure they are running as expected.
- Keep track of successful backups, failed backups, and any warnings or errors.
- This monitoring can be done manually or by leveraging monitoring tools or dashboards provided by the backup solution.
- **Test backup recovery:** Periodically test the backup recovery process to verify the integrity and reliability of the backup data.
- Perform test restores to ensure that backups can be successfully restored and that the data is accessible and usable.
- Regular testing helps identify any potential issues or gaps in the backup process.

SYSTEM LOGS, UPDATES, PATCHES, AND CONFIGURATION CHANGES, BACKUPS..

- **Secure backup storage:** Ensure that the backup data is securely stored in a separate location from the production system.
- Consider encrypting the backup data to protect it from unauthorized access.
- If using cloud-based backup services, review their security measures and ensure they align with your requirements.
- **Document the backup process:** Maintain clear documentation of the backup process, including the automation scripts, schedules, and any configurations or customizations.
- This documentation serves as a reference for future administrators and helps ensure consistency in the backup process.

INSTALLING AND CONFIGURING HARDWARE AND SOFTWARE

- The process of installing and configuring hardware and software involves setting up and managing the necessary components to support the software system.
- Here's an overview of how this process typically unfolds in software administration:
- **Hardware installation:** Software administrators may be responsible for installing hardware components that are essential for the software system to operate effectively.
- This can include servers, network devices, storage systems, and other infrastructure elements.
- The installation process involves physically setting up the hardware, connecting cables and peripherals, and ensuring proper power supply and cooling.

INSTALLING AND CONFIGURING HARDWARE AND SOFTWARE

- **Operating system installation:** Software administrators are often involved in installing and configuring the operating system (OS) that will serve as the foundation for running the software system.
- They typically follow the OS vendor's installation instructions, which may involve booting from installation media, partitioning disks, selecting installation options, and providing necessary settings such as time zones, network configurations, and administrative accounts.
- **Dependency installation:** Software systems often have dependencies on additional software components, libraries, or frameworks.
- This may involve installing database management systems, web servers, application servers, programming language runtimes, or other required software components.

INSTALLING AND CONFIGURING HARDWARE AND SOFTWARE

- **Software installation:** Software administrators handle the installation of the actual software system they are managing.
- This typically involves running the installation program or package provided by the software vendor.
- During the installation, administrators may need to specify installation paths, configure database connections, set up administrative accounts, and provide any required license keys or activation information.

INSTALLING AND CONFIGURING HARDWARE AND SOFTWARE

- **Configuration and customization:** After the software installation, software administrators configure the system to align with the specific requirements of the organization or users.
- This includes setting up user accounts, defining security settings, configuring network connectivity, adjusting performance parameters, and customizing the software's behavior to meet desired specifications.
- Configuration may involve modifying configuration files, using administrative consoles or interfaces provided by the software, or utilizing command-line tools.

INSTALLING AND CONFIGURING HARDWARE AND SOFTWARE

- **Integration with other systems:** it involve setting up communication protocols, establishing data exchange formats, configuring APIs or web services, or implementing authentication and authorization mechanisms.
- **Testing and validation:** Once the hardware and software components are installed and configured, software administrators perform testing and validation to ensure that the system functions as intended.
- It includes conducting functional tests, performance tests, security assessments, and verifying system behavior against defined requirements.
- **Ongoing maintenance and updates:** Software administrators are responsible for managing ongoing maintenance tasks, such as applying software updates, security patches, and bug fixes.
- They also monitor system performance, troubleshoot issues, and ensure system availability and reliability over time.

INSTALLING AND CONFIGURING HARDWARE AND SOFTWARE

- Hardware Installation Best Practices:
- **Plan and prepare:** Before installing hardware, carefully plan and evaluate the requirements, compatibility, and scalability of the system.
  - Ensure that you have all the necessary components, cables, and accessories.
- **Follow manufacturer guidelines:** Always refer to the manufacturer's documentation and guidelines for hardware installation.
  - Adhere to their recommended procedures, safety precautions, and any specific requirements they outline.
- **Proper physical setup:** Pay attention to proper grounding, cable management, and ventilation to prevent overheating and ensure optimal performance and reliability.

INSTALLING AND CONFIGURING HARDWARE AND SOFTWARE

- Hardware Installation Best Practices:

- **Label and document:** Label hardware components and document their location, specifications, and any unique identifiers.
  - This helps with future maintenance, troubleshooting, and component identification.

- **Regular maintenance:** Establish a schedule for regular hardware maintenance, including cleaning, inspecting cables and connections, and checking for any signs of wear or damage.
  - Regular maintenance helps prevent potential hardware failures and ensures system stability.

INSTALLING AND CONFIGURING HARDWARE AND SOFTWARE

- Software Installation and Configuration Best Practices:

- **Read documentation:** Thoroughly review the software vendor's documentation, user guides, and release notes.
  - Understand the installation and configuration requirements, dependencies, and best practices recommended by the vendor.

- **System compatibility:** Ensure that the software is compatible with the underlying hardware, operating system, and other software components.
  - Verify the system requirements and compatibility matrix provided by the software vendor.

INSTALLING AND CONFIGURING HARDWARE AND SOFTWARE

Software Installation and Configuration Best Practices..

- **Secure installation:** Apply security best practices during software installation, such as using strong passwords, enabling encryption where applicable, and configuring proper access controls.
- **Centralized management:** Utilize centralized management tools or configuration management systems to streamline software installation and configuration across multiple systems.
- It helps maintain consistency and simplifies administration tasks.
- **Standardize configurations:** Establish standardized configurations and settings for software installations.
- This ensures consistency, simplifies troubleshooting, and facilitates scalability and maintenance.

INSTALLING AND CONFIGURING HARDWARE AND SOFTWARE

- **Regular updates and patches:** Stay up-to-date with software updates, patches, and security fixes provided by the vendor.
- Regularly apply these updates to mitigate security vulnerabilities, improve performance, and access new features.
- **Testing and validation:** Thoroughly test the installed software and its configurations to ensure they function as expected.
- Conduct functional tests, performance tests, and security assessments to validate the system's behavior and identify any issues.
- **Backup and recovery:** Implement a robust backup and recovery strategy for the software system.
- Regularly back up configurations, databases, and critical data to ensure data integrity and facilitate disaster recovery if needed.

INSTALLING AND CONFIGURING HARDWARE AND SOFTWARE

- Software Installation and Configuration Best Practices:
- **Document changes and configurations:** Maintain detailed documentation of software installations, configurations, and any changes made.
- This documentation serves as a reference for future troubleshooting, upgrades, and system maintenance.
- **Training and knowledge sharing:** Ensure that software administrators receive proper training and knowledge sharing sessions to stay updated with the latest installation and configuration best practices.
- Encourage collaboration and knowledge sharing within the team to foster expertise and improve overall efficiency.

SYSTEM PERFORMANCE TUNING

- System performance tuning is an important aspect of software administration that involves optimizing the performance of a computer system to ensure efficient and smooth operation.

- some key areas to focus on when tuning system performance:

- **1. Resource Monitoring and Analysis:**

- Monitor system resources such as CPU usage, memory utilization, disk I/O, and network activity using tools like top, htop, or system monitoring utilities.

- **2. System Configuration:**

- Review and optimize system configuration settings, such as kernel parameters, network settings, and file system parameters.

- Adjust parameters related to process scheduling, memory allocation, disk I/O, and network buffers to match the workload and hardware characteristics of the system.

SYSTEM PERFORMANCE TUNING…

3. Disk I/O Optimization:

- Optimize disk I/O performance by using techniques such as RAID configurations, disk partitioning, optimizing file system parameters, and utilizing solid-state drives (SSDs) or disk caching mechanisms.
- Use tools like iotop to monitor disk I/O activity and identify any processes causing excessive disk I/O.

4. Memory Management:

- Ensure efficient memory management by adjusting settings related to memory allocation, swapping, and caching.
- Optimize memory usage by tuning application settings, reducing unnecessary memory consumption, and using memory profiling tools to identify memory leaks or excessive memory usage.

SYSTEM PERFORMANCE TUNING...

5. CPU and Process Optimization:

- Monitor CPU utilization and identify processes consuming high CPU resources.
- Use tools like top or htop to identify CPU-intensive processes.
- Consider optimizing CPU-bound processes by parallelizing computations, optimizing algorithms, or distributing workloads across multiple CPUs or cores.

6. Network Optimization:

- Optimize network performance by tuning network settings, adjusting buffer sizes, and optimizing network protocols.
- Identify and resolve network bottlenecks, such as congested network interfaces or misconfigured network devices.

SYSTEM PERFORMANCE TUNING…

7. Application-Level Optimization:

- Optimize software applications by reviewing and fine-tuning application-specific settings, database queries, and caching mechanisms.
- Profile application performance using tools like strace, perf, or application-specific profiling tools to identify performance bottlenecks and areas for improvement.

8. Regular Maintenance and Updates:

- Keep the system up to date with the latest software updates, security patches, and bug fixes.
- Regularly clean up unnecessary files, logs, and temporary data to free up disk space and improve overall system performance.

SYSTEM PERFORMANCE TUNING…

- Tools and utilities available for monitoring system resources in various operating systems.

1. **top/htop:** top and htop are command-line utilities available in many Unix-based systems.

- provide real-time monitoring of system resource usage, including CPU, memory, and process information.
- allow sorting processes based on different criteria such as CPU usage or memory usage.

- **2.vmstat:** vmstat is a command-line utility that provides detailed information about system virtual memory, CPU usage, disk I/O, and other system statistics.
- It can be used to monitor overall system performance and identify bottlenecks.

- **3.iostat:** iostat is a command-line utility that provides disk I/O statistics, including disk utilization, throughput, and average response time.

SYSTEM PERFORMANCE TUNING…

- **4. sar:** sar (System Activity Reporter) is a command-line utility that collects and reports system activity and resource usage over time.
- It provides a wealth of information, including CPU utilization, memory usage, disk I/O, and network activity.
- It can generate reports and help analyze system performance trends.
- **5. nmon**: nmon is a command-line monitoring tool available for various Unix-based systems.
- It provides a comprehensive view of system resources, including CPU, memory, disk I/O, network, and more.
- It displays real-time data in a user-friendly format and can generate reports for further analysis.
- **6. Windows Task Manager:** Task Manager is a built-in utility in Windows operating systems.
- It provides real-time monitoring of CPU, memory, disk usage, and network activity.

SYSTEM PERFORMANCE TUNING…

- **7. Resource Monitor:** is built-in utility in Windows that provides detailed information about system resource usage.
- It monitors CPU, memory, disk, and network activity and allows sorting processes based on various criteria.

- **8. Performance Monitor:** Performance Monitor (perfmon or perfmon.exe) is a Windows utility for advanced monitoring and analysis of system performance.
- It offers a wide range of performance counters, including CPU usage, memory utilization, disk activity, network performance, and more.
- It allows creating custom monitoring setups and generating reports.

PERFORMING ROUTINE AUDITS OF SYSTEMS AND SOFTWARE.

- Performing routine audits of systems and software is an important practice to ensure the security, stability, and compliance of your IT infrastructure.
- **Steps you can follow to conduct routine audits:**
- **Define Audit Objectives:** Determine the goals and scope of the audit. Identify the systems, software, and areas of focus that need to be audited.
- Common areas include security controls, access management, configuration settings, software versions, and compliance requirements.
- **Establish Audit Criteria:** Define the criteria against which you will evaluate the systems and software.
- This includes industry best practices, security standards (e.g., ISO 27001), regulatory requirements (e.g., GDPR, HIPAA), and internal policies and procedures.

PERFORMING ROUTINE AUDITS OF SYSTEMS AND SOFTWARE.

- **Perform Security Assessments:** Conduct vulnerability assessments and penetration testing to identify security weaknesses in systems and software.
- Use automated tools and manual techniques to assess vulnerabilities, misconfigurations, and potential entry points for unauthorized access.
- **Review Access Controls:** Evaluate user access controls and permissions for systems and software.
- Ensure that access rights are granted based on the principle of least privilege, and regularly review and remove unnecessary or outdated user accounts.
- **Check Configuration Settings:** Review system and software configuration settings to ensure they align with security best practices and compliance requirements.
- Check for default or weak configurations, unauthorized changes, and potential vulnerabilities.

PERFORMING ROUTINE AUDITS OF SYSTEMS AND SOFTWARE.

- **Assess Patch Management:** Evaluate the patch management process for systems and software.
- Verify that critical security patches and updates are being applied in a timely manner to mitigate known vulnerabilities.
- **Review Logs and Monitoring:** Analyze system logs, event logs, and security monitoring data to identify any suspicious activities or security incidents.
- Ensure that appropriate logging and monitoring mechanisms are in place and are functioning effectively.
- **Evaluate Backup and Disaster Recovery:** Assess backup procedures and disaster recovery plans for systems and software.
- Verify that regular backups are performed, test the restoration process, and ensure the backups are securely stored.

PERFORMING ROUTINE AUDITS OF SYSTEMS AND SOFTWARE.

- **Check Compliance Requirements:** Review systems and software against applicable compliance requirements such as GDPR, HIPAA, PCI DSS, or industry-specific regulations.
- Ensure that data handling, privacy, and security practices are in line with the required standards.
- **Document Findings and Remediate:** Document the audit findings, including identified vulnerabilities, issues, and areas for improvement.
- Develop a remediation plan to address the identified issues and implement necessary changes or fixes.
- **Follow-Up and Periodic Audits:** Monitor the progress of remediation efforts and perform periodic audits to ensure that the identified issues are resolved and that systems and software remain secure and compliant over time.

HOW OFTEN SHOULD ORGANIZATIONS CONDUCT ROUTINE AUDITS?

- The frequency of routine audits for systems and software can vary depending on factors such as the size and complexity of the organization, industry regulations, and the level of risk associated with the systems.
- General guidelines for conducting routine audits:
- **Regular Intervals:** Conducting audits at regular intervals helps ensure ongoing monitoring and security of systems and software.
- Quarterly or bi-annual audits are common frequencies for many organizations.
- This allows for timely identification of vulnerabilities and compliance gaps.
- **Significant Changes:** Perform audits when significant changes occur in the IT infrastructure.
- It includes major system upgrades, implementation of new software, changes in network architecture, or modifications to security controls.

HOW OFTEN SHOULD ORGANIZATIONS CONDUCT ROUTINE AUDITS?

- **Incident Response Triggers:** If a security incident occurs, conduct an audit to identify the root cause, assess the impact, and strengthen security controls.
- **Vendor Software Updates:** When software vendors release major updates or security patches, it is advisable to conduct audits to ensure the timely application of patches, assess the impact of the updates on the system, and verify continued compliance.
- **Regulatory Changes:** Keep track of any changes in industry regulations and compliance requirements.
- Conduct audits as necessary to align with updated standards and ensure continued compliance.

HOW OFTEN SHOULD ORGANIZATIONS CONDUCT ROUTINE AUDITS?

- **Compliance Requirements:** Organizations subject to specific compliance regulations should align their audit frequency with the mandated requirements. Industries such as healthcare (HIPAA), finance (PCI DSS), or government contracts (NIST, FedRAMP) often have specific audit frequency requirements.
- **Risk Assessment:** Conduct audits based on risk assessments. Evaluate the criticality and sensitivity of the systems and software, and prioritize audits for high-risk areas. High-value systems, sensitive data repositories, or systems with a history of security incidents may require more frequent audits.

Discussion..