Ocean Graham

Abstract

[Draw your reader in with an engaging abstract. It is typically a short summary of the document.   
When you’re ready to add your content, just click here and start typing.]

401IT – Operating Systems

[Assignment Title]



Table of Contents

[Table of Figures 1](#_Toc202859948)

[Introduction 2](#_Toc202859949)

[Operational Management of Operating Systems 2](#_Toc202859950)

[Process and service management 2](#_Toc202859951)

[file systems 7](#_Toc202859952)

[user accounts and access control 7](#_Toc202859953)

[memory management 7](#_Toc202859954)

[Security risks and management strategies 8](#_Toc202859955)

[OS vulnerabilities 8](#_Toc202859956)

[management strategies 8](#_Toc202859957)

[References 8](#_Toc202859958)

# Table of Figures

401IT – Operating Systems

# Introduction

In this coursework the author will demonstrate a high degree of understanding of operating systems and a critical analysis of the various operational aspects. This report will evaluate different aspects in relation to different operating systems, these being Linux, Windows and Windows Server.

# Operational Management of Operating Systems

## Process and service management

There are two types of functions that run on a computer, these being ‘Processes’ and ‘Services’. We first must identify the definition and differences between these two to understand how they are managed.

A Process is a program that requires initiation manually and is unnecessary to the function of the system. Processes are made of attributes to help the OS manage and control it. These being stored in the Process Control Block (PCB). The PCB stores these attributes:

1. Process States
2. Process ID
3. Priority levels
4. Input Output Information
5. File Descriptors
6. Account Information
7. Memory Management Information

(GeeksforGeeks, 2024)

The PCB help the system share resources more effectively, it also allows the system to isolate processes, meaning ‘one process doesn’t inadvertently interfere with another’. (Vaishnav, 2022)

Processes may vary on the different operating systems but some examples for the different systems may be:

Linux –

* bash (command line)
* python (python script runner)
* vim (text editor)
* Search Engines

A screenshot of a computer program

AI-generated content may be incorrect.

Figure – Linux Task Manager - Processes

Windows –

* Windows Explorer (file explorer)
* Command Prompt (command line)
* Virtual Studio Code (script editor and runner)
* Word (text editor)
* Search Engines

A screenshot of a computer

AI-generated content may be incorrect.

Figure – Windows Task Manager - Processes

A service is a background process that performs necessary functions, managing hardware, all without user interaction. Services initiate automatically with the startup of the operating system. These manage many things such as:

* Program Execution
* Input/Output Operations
* Memory Management
* Process Management
* File System Manipulation

(*Operating System - Services - Tutorialspoint*, 2019)

Examples of Services include:

* Print Spooler
* Radio Management Services
* Windows Update Services (Windows)
* Apache2 (Linux)

A screenshot of a computer

AI-generated content may be incorrect.

Figure – Windows Services Page and Print Spooler Service

A screenshot of a computer

AI-generated content may be incorrect.

Figure – Windows Server Service Page

Processes and Services can be handled and managed using both Graphical User Interfaces (GUI) and Command-Line Interfaces (CLI). The 3 prior figures provide examples of GUI management of the functions. These two interfaces provide different strengths and weaknesses in use, the GUI providing easier user experience and effects to make it easier to navigate, whilst CLI is rawer and more direct. It is much more efficient and faster than the GUI and requires less resources (Marijan, 2023). The speed of the CLI gives value to admin users for troubleshooting.

A screenshot of a computer program

AI-generated content may be incorrect.

Figure – Windows CLI showing running processes

A computer screen shot of white text

AI-generated content may be incorrect.

Figure – Windows CLI showing service information

A screenshot of a computer

AI-generated content may be incorrect.

Figure – Linux CLI showing service information

Figure x, x and x show the CLI and how they may be used to find and look at process and service information in an alternative way to the GUI.

## file systems

A file system is a logical and physical system for organising, managing and accessing the files and directories on a devices storage media (Sullivan, 2018). Without a file system the system would have large amounts of data without any organisation or distinguishment. File systems between different OS have different approaches to file storage and therefore different architectures.

Linux uses a file system designed specifically for Linux in 1992. The Extended File System has multiple iterations, but currently is on its 4th iteration EXT4 which

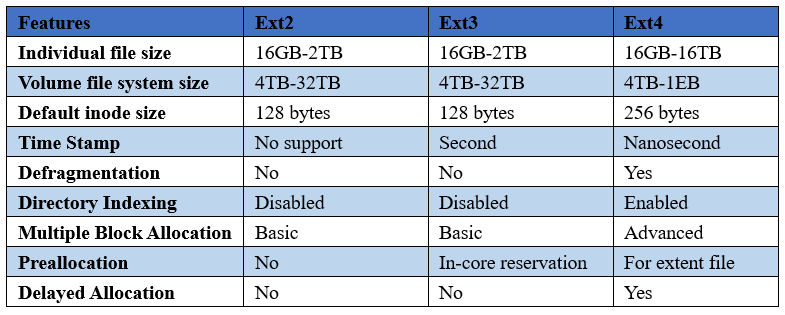
improves file and volume storage and contain other features such as Defragmentation and Journaling. On the other hand, EXT4 requires more disk usage compared to the older versions however this is to be expected with its higher complexity.

Figure – Comparison of iterations of EXT (*Partition Wizard,* 2021)

Windows uses multiple different systems, the File Allocation Table (FAT) and New Technology File System (NTFS)

Windows FAT NTFS

Windows server ReFS

Explain the structure and management of file systems in Linux, Windows, and Windows Server.

Describe the hierarchical organization of files and directories in these operating systems.

Discuss the importance of file system permissions in securing data.

Explain how file system permissions are managed differently across Linux, Windows, and Windows Server.

Provide an overview of command-line and GUI methods to create, modify, and delete files and directories

Compare FAT32, NTFS, ext4

Features: performance, security, compatibility

Show how folders/files are created or managed

## user accounts and access control

Describe the principles of user account management across different operating systems.

Explain the significance of user accounts in system security and resource management.

Explain the process of creating, modifying, and deleting user accounts in Linux, Windows, and Windows Server.

Discuss how access control is implemented using Access Control Lists (ACLs) and Role-Based Access Control (RBAC) in these systems.

Highlight the importance of access control in preventing unauthorized access and ensuring system integrity.

Types of user accounts (admin, standard, guest)

Permissions, groups, access rights

Password and account security

## memory management

Describe the principles of memory management in various operating systems.

Cover key concepts such as stack, heap, shared memory, virtual memory, addressing, paging, swapping, buffers, and ring buffers

Explain how different operating systems handle memory management tasks.

Discuss the significance of efficient memory management in system performance and stability.

Compare and contrast the memory management techniques used by Linux, Windows, and Windows Server.

Virtual memory, paging, RAM usage

Swap files/page files

Memory limits per user/app

# Security risks and management strategies

a

## OS vulnerabilities

a

## management strategies

a

# References