

OPERATING SYSTEMS REPORT

STUDENT NAME: Zacharias Lambrou

STUDENT ID: 2435234

CAMPUS: University Of Wales Trinity Saint David (London Campus)

MODULE: Computer Architecture And Operating Systems

LECTURER: Dr Redhwan Al-amri

Group: 2

Contents

Introduction	2
1 Key Functions of General-Purpose Operating Systems	2
Process Management	2
Memory Management	2
File and Directory Management	2
Device Management and Additional Services	3
Comparative Analysis	3
2. Use of Windows OS CLI Commands	7
Process management	7
File and Directory Management	9
Network Testing and Monitoring	11
User Management.....	14
Task Scheduling.....	15
3. Use of Linux OS CLI Commands	17
Process management	17
File & directory management.....	18
Network testing/monitoring.....	18
User management	19
4. Use of Windows OS Scripts	21
5. Use of Linux OS Scripts	24
Conclusion	26
Reference List	27
Appendices	28
Appendix 3.....	28

Introduction

This report explores the main functions of popular general-purpose operating systems and demonstrates practical command-line usage in both Windows and Linux. It also shows how simple scripts can automate everyday tasks. All commands were run on my own laptop (Windows 11) and Ubuntu virtual machine, with screenshots taken directly from my screen as evidence. Multiple sources were used to support the explanations.

1 Key Functions of General-Purpose Operating Systems

Operating systems act as the central control mechanism for computer systems, facilitating the seamless integration of hardware and software components to ensure stable and efficient operation. Prominent general-purpose operating systems include Microsoft Windows, various Linux distributions such as Ubuntu, and Apple's macOS. Although these platforms share fundamental responsibilities, they differ significantly in their user interfaces and design philosophies [1], [2]. The core functions of an operating system encompass process management, memory management, file and directory management, device management, networking, security, and user interface services.

Process Management

Process management involves the initiation, scheduling, monitoring, and termination of concurrently executing programs. The operating system allocates CPU time slices to each process to prevent resource contention and system hangs. In Windows, the Task Manager provides real-time visibility into active processes, displaying CPU and memory utilisation and enabling the termination of unresponsive tasks. Linux achieves similar oversight through command-line utilities such as *ps* and *top*, as well as graphical system monitors. Inadequate process management can result in system freezes due to competition for CPU resources [3], [4].

Memory Management

Memory management is responsible for allocating Random Access Memory (RAM) to individual programs and employs virtual memory techniques, such as swapping data to secondary storage when physical RAM is insufficient. This mechanism prevents memory conflicts between applications. Windows provides Resource Monitor and the Performance tab in Task Manager for detailed memory monitoring, while macOS offers the Activity Monitor with comparable functionality. Linux's open-source kernel allows extensive optimisation, making it particularly effective in server environments where performance and stability are critical [2], [5].

File and Directory Management

File and directory management organises data into hierarchical structures, enforces access permissions to protect data integrity, and prevents corruption. Windows utilises File Explorer for this purpose, whereas Linux offers both graphical file managers and powerful command-line tools such as *ls*, *cp*, and

mv. Both systems support the viewing of hidden files and extend to drive formatting and storage space monitoring [1], [6].

Device Management and Additional Services

Device management enables communication between the operating system and hardware peripherals, including printers, USB devices, and graphics cards, using appropriate drivers. Integrated security features such as firewalls, user account management, and encryption are essential. Networking capabilities support connectivity to the internet and other devices, while the user interface — encompassing the Windows Start menu, Linux desktop environments (e.g., GNOME), and the macOS dock — provides the primary means of interaction [3], [7]. Contemporary operating systems also incorporate automated power management and update mechanisms [4], [5].

Comparative Analysis

Despite their varied appearances, modern general-purpose operating systems perform a consistent set of core functions. Windows remains popular for general computing and gaming, Linux excels in server environments due to its customisability and stability, and macOS delivers a streamlined experience within the Apple ecosystem. Cross-referencing information from multiple sources has been invaluable in understanding both the underlying similarities and the platform-specific strengths of these operating systems [2], [6]

Task Manager

Type a name, publisher, or PID to search

Processes

Name	Status	16% CPU	82% Memory	4% Disk	0% Network
Apps (5)					
Google Chrome (17)		0%	468.3 MB	0.1 MB/s	0 Mbps
Microsoft Word (2)		0%	108.2 MB	0 MB/s	0 Mbps
Task Manager		1.6%	74.4 MB	0.1 MB/s	0 Mbps
Terminal (3)		0%	6.4 MB	0 MB/s	0 Mbps
Webex		0%	3.6 MB	0.1 MB/s	0 Mbps
Background processes (111)					
Acrobat Collaboration Synchr...		0%	9.6 MB	0 MB/s	0 Mbps
Acrobat Update Service (32 bit)		0%	0.4 MB	0 MB/s	0 Mbps
Adobe Content Synchronizer (...)		0%	4.4 MB	0 MB/s	0 Mbps
Adobe Express Photos (5)		0.4%	10.2 MB	0 MB/s	0 Mbps
AMD Crash Defender Service		0%	0.5 MB	0 MB/s	0 Mbps
AMD External Events Client M...		0%	1.1 MB	0 MB/s	0 Mbps
AMD External Events Service ...		0%	0.5 MB	0 MB/s	0 Mbps
AMD Software (4)		0%	11.0 MB	0 MB/s	0 Mbps
Antimalware Core Service		0%	3.7 MB	0 MB/s	0 Mbps
Antimalware Service Executable		0%	148.5 MB	0.2 MB/s	0 Mbps
App Actions		0%	2.5 MB	0 MB/s	0 Mbps
Application Frame Host		0%	1.0 MB	0 MB/s	0 Mbps
AutoCorrect (32 bit)		0%	42.6 MB	0 MB/s	0 Mbps
Browser (5)		0%	14.8 MB	0 MB/s	0 Mbps
ChatGPT (5)		0%	32.7 MB	1.5 MB/s	0 Mbps
Cisco Webex Meetings (32 bit)		0%	1.0 MB	0 MB/s	0 Mbps

Fig.1 Task Manager

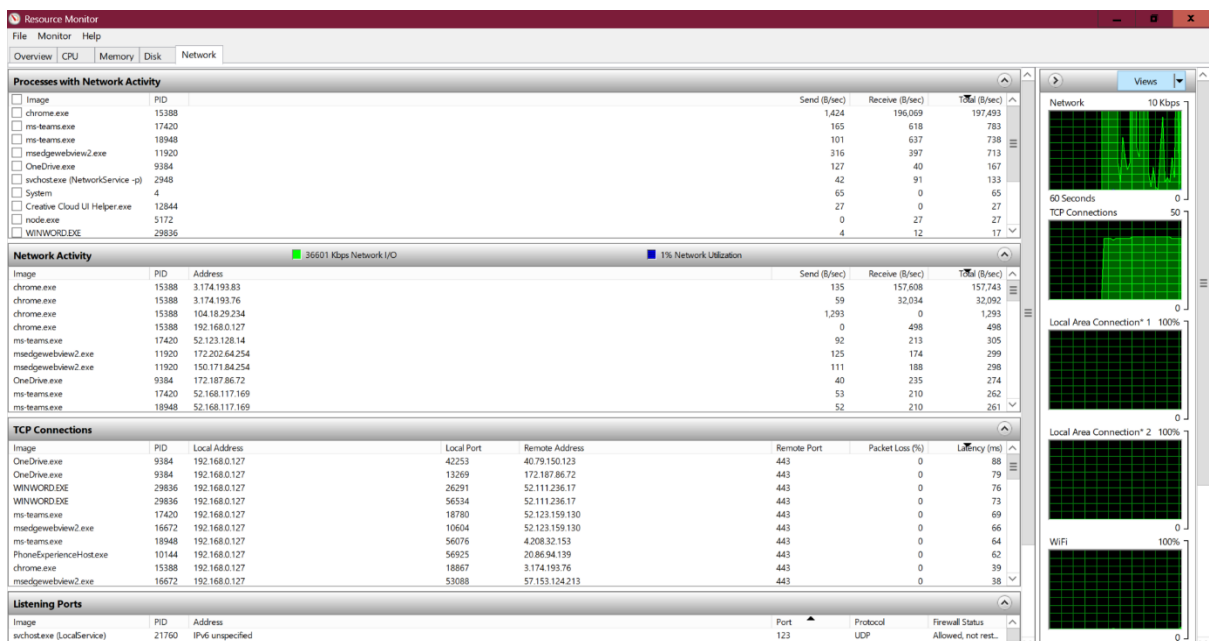


Fig. 2: Resource Monitor

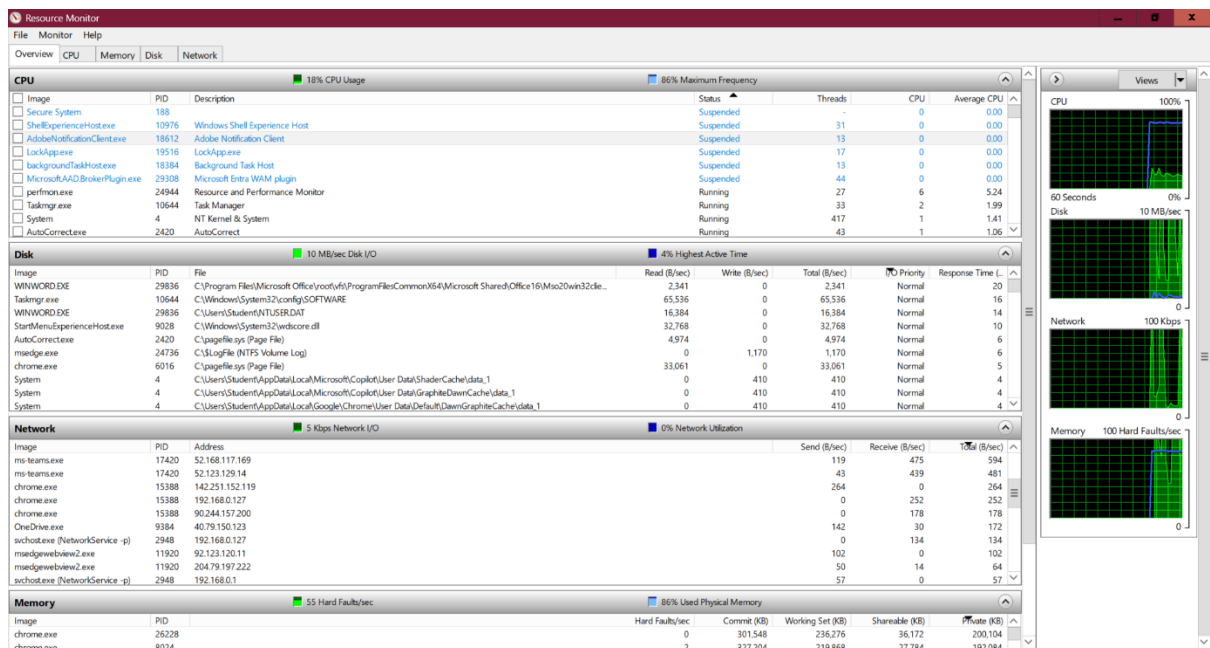


Fig. 3: Windows Resource Monitor

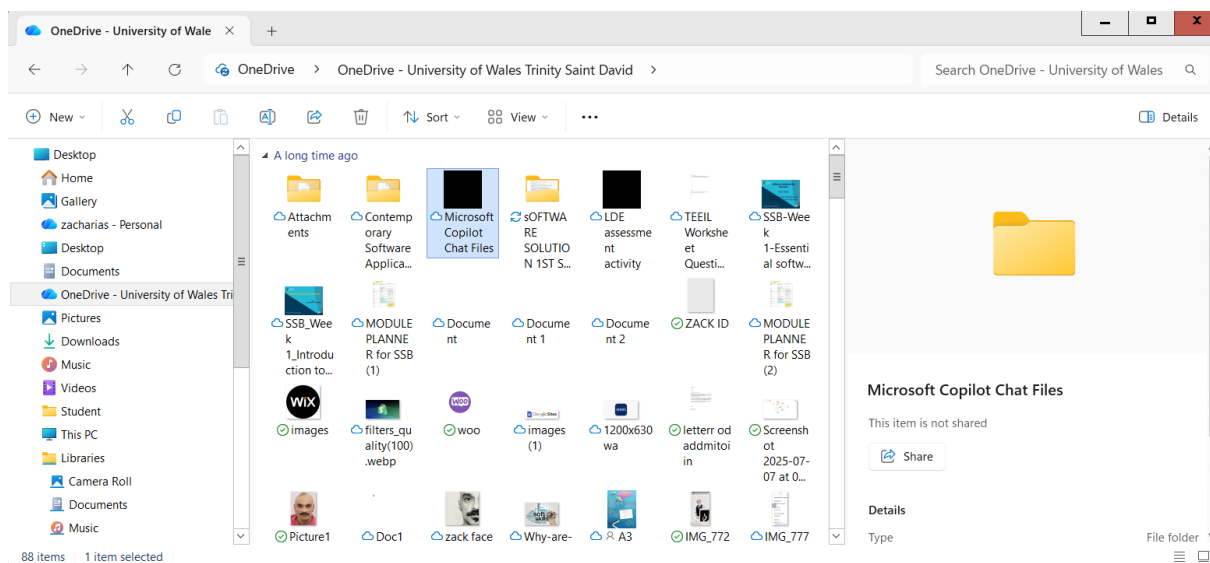


Fig. 4: Windows File Explorer

```

top - 02:59:17 up 6 days, 6:07, 1 user, load average: 0.30, 0.35, 0.42
Tasks: 8 total, 1 running, 7 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1.1 us, 0.5 sy, 0.0 ni, 98.1 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st
KiB Mem: 65849428 total, 21359740 used, 44489688 free, 280028 buffers
KiB Swap: 0 total, 0 used, 0 free. 10663652 cached Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM     TIME+ COMMAND
    1 root        20   0   17984   2944   2696  S   0.0   0.0   0:00.07 init
   32 syslog     20   0  182132   2952   2444  S   0.0   0.0   0:00.01 rsyslogd
   36 root        20   0   23676   2120   1900  S   0.0   0.0   0:00.00 cron
   38 root        20   0   61404   3120   2440  S   0.0   0.0   0:00.00 sshd
   55 bind       20   0  892824  40024   5100  S   0.0   0.1   0:00.13 named
   76 root        20   0   63156   2824   2368  S   0.0   0.0   0:00.00 login
   86 sysadmin   20   0   18200   3372   2884  S   0.0   0.0   0:00.01 bash
   97 sysadmin    20   0   19876   2416   2088  R   0.0   0.0   0:00.00 top

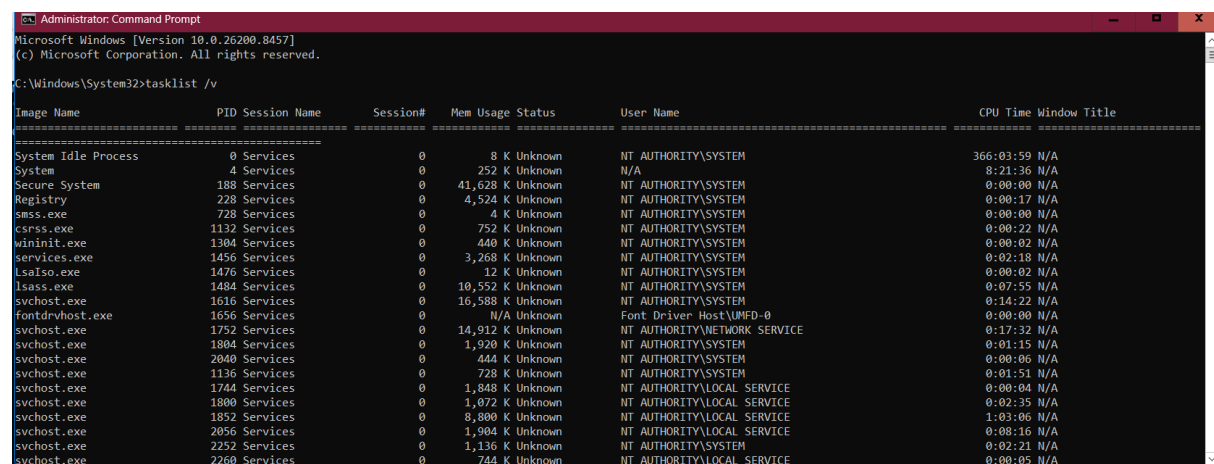
```

Fig. 5: Linux top

2. Use of Windows OS CLI Commands

Command Prompt and PowerShell were used to explore Windows command-line functionality on a Windows 11 laptop. The following screenshots are from different commands, incorporating relevant switches and options, were executed to demonstrate practical system administration across key areas. Screenshots were captured showing both the command syntax and resulting output [8], [9].

Process management

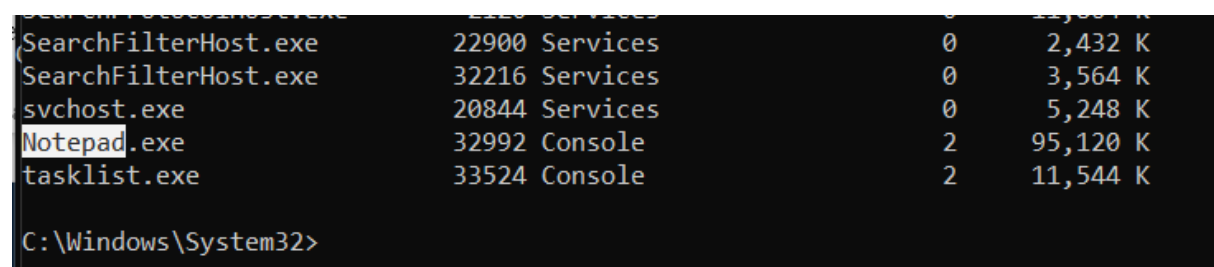


```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.26200.8457]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>tasklist /v

Image Name                PID Session Name        Session#    Mem Usage Status      User Name                CPU Time Window Title
=====
System Idle Process        0 Services              0           8 K Unknown    NT AUTHORITY\SYSTEM      366:03:59 N/A
System                     4 Services              0          252 K Unknown    N/A                      8:21:36 N/A
Secure System             188 Services            0       41,628 K Unknown    NT AUTHORITY\SYSTEM      0:00:00 N/A
Registry                  228 Services            0       4,524 K Unknown    NT AUTHORITY\SYSTEM      0:00:17 N/A
smss.exe                   728 Services            0           4 K Unknown    NT AUTHORITY\SYSTEM      0:00:00 N/A
csrss.exe                  1132 Services           0          752 K Unknown    NT AUTHORITY\SYSTEM      0:00:22 N/A
wininit.exe               1304 Services            0          440 K Unknown    NT AUTHORITY\SYSTEM      0:00:02 N/A
services.exe              1456 Services            0       3,268 K Unknown    NT AUTHORITY\SYSTEM      0:02:18 N/A
lsass.exe                 1476 Services            0          12 K Unknown    NT AUTHORITY\SYSTEM      0:00:02 N/A
lsass.exe                 1484 Services            0      10,552 K Unknown    NT AUTHORITY\SYSTEM      0:07:55 N/A
svchost.exe               1616 Services            0      16,588 K Unknown    NT AUTHORITY\SYSTEM      0:14:22 N/A
fontdrvhost.exe          1656 Services            0           N/A Unknown    Font Driver Host\UMFD-0  0:00:00 N/A
svchost.exe               1752 Services            0      14,912 K Unknown    NT AUTHORITY\NETWORK SERVICE 0:17:32 N/A
svchost.exe               1804 Services            0          1,920 K Unknown    NT AUTHORITY\SYSTEM      0:01:15 N/A
svchost.exe               2040 Services            0          444 K Unknown    NT AUTHORITY\SYSTEM      0:00:06 N/A
svchost.exe               1136 Services            0          728 K Unknown    NT AUTHORITY\SYSTEM      0:01:51 N/A
svchost.exe               1744 Services            0       1,848 K Unknown    NT AUTHORITY\LOCAL SERVICE 0:00:04 N/A
svchost.exe               1800 Services            0       1,072 K Unknown    NT AUTHORITY\LOCAL SERVICE 0:02:35 N/A
svchost.exe               1852 Services            0       8,800 K Unknown    NT AUTHORITY\LOCAL SERVICE 1:03:06 N/A
svchost.exe               2056 Services            0       1,904 K Unknown    NT AUTHORITY\LOCAL SERVICE 0:08:16 N/A
svchost.exe               2252 Services            0       1,136 K Unknown    NT AUTHORITY\SYSTEM      0:02:21 N/A
svchost.exe               2260 Services            0          744 K Unknown    NT AUTHORITY\LOCAL SERVICE 0:00:05 N/A
```

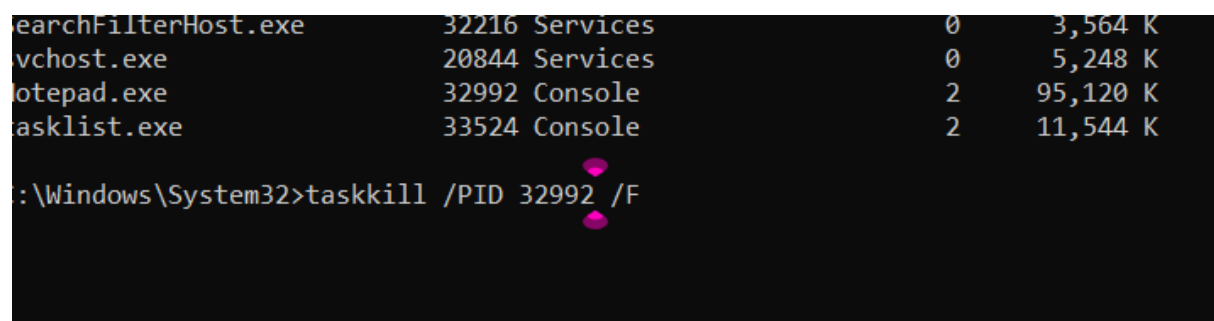
Fig. 6 : tasklist /v – lists all running processes with extra info like memory and status



```
SearchFilterHost.exe      22900 Services              0       2,432 K
SearchFilterHost.exe      32216 Services              0       3,564 K
svchost.exe               20844 Services              0       5,248 K
Notepad.exe               32992 Console                 2      95,120 K
tasklist.exe              33524 Console                 2     11,544 K

C:\Windows\System32>
```

Fig. 7: The Notepad in tasklist



```
SearchFilterHost.exe      32216 Services              0       3,564 K
svchost.exe               20844 Services              0       5,248 K
Notepad.exe               32992 Console                 2      95,120 K
tasklist.exe              33524 Console                 2     11,544 K

C:\Windows\System32>taskkill /PID 32992 /f
```

Fig. 8: taskkill /PID 32992 /f – forces a process to close by its PID. I used it on a test notepad.

```
C:\Windows\System32>taskkill /PID 32992 /F
SUCCESS: The process with PID 32992 has been terminated.
C:\Windows\System32>
```

Fig. 9: The process with PID 32992 (Notepad) had been terminated

I found tasklist and taskkill useful for cleaning up stuck programs. The network ones helped me check why my Wi-Fi was slow once. The commands that I used worked as expected when I typed them carefully. I used a few options/switches like /f for force or -n for count because the basic command alone doesn't give enough detail [9], [10].

File and Directory Management

Commands such as `dir /s /b` provided recursive file listings in bare format, while `del /q /s *.tmp` enabled silent deletion of temporary files across directories. These operations illustrate efficient file system navigation and maintenance directly from the command line.

```
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\mail merge example.docx
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\MAIL MERGE Receipien
ce.docx
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Microsoft_Access-Log
o.wine (1).png
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Microsoft_Access-Log
o.wine.png
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Microsoft_Excel-Logo
.wine.png
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Microsoft_PowerPoint
-Logo.wine.png
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Microsoft_Teams-Logo
.wine.png
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Microsoft_Word-Logo.
wine.png
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Presentation.pptx
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\screen shot.docx
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Shopify-Logo.wine.pn
g
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Software Solution fi
nal final 2 (2.docx
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Software Solution fi
nal final 2 .docx
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Software Solution fi
nal final submission 2.pdf
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Spotify-Logo.wine.pn
g
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\SSB Portfolio guidel
ine Template and marking Scheme.docx
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\student email practi
cve .xlsx
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\Wix.com-Logo.wine.pn
g
C:\Users\Student\OneDrive - University of Wales Trinity Saint David\sOFTWARE SOLUTION 1ST SEMEZSTER\WooCommerce-Logo.win
e.png
C:\Users\Student\Searches\winrt--{S-1-5-21-572943906-1319465739-1045318031-1001}-.searchconnector-ms
C:\Users\Student\Videos\Captures
C:\Users\Student\Videos\database exams walkthrough.mp4

C:\Users\Student>dir /s /b
```

Fig. 10: `dir /s /b` – shows all files recursively in bare format

```

.\Users\Student> del /q /s *.tmp
Deleted file - C:\Users\Student\AppData\Local\CiscoSpark\2dbf2fcc-13b9-4c34-8d7d-b837545fe598\temp_index_3_v2.tmp
Deleted file - C:\Users\Student\AppData\Local\CiscoSpark\8317110274882156666\temp_index_1_v2.tmp
Deleted file - C:\Users\Student\AppData\Local\CiscoSpark\8317110274882156666\temp_index_4_v2.tmp
Deleted file - C:\Users\Student\AppData\Local\CiscoSpark\8317110274882156666\temp_index_v2.tmp
Deleted file - C:\Users\Student\AppData\Local\CiscoSpark\Logs\accessories\aci_5880_115_temp_index_v2.tmp
Deleted file - C:\Users\Student\AppData\Local\CiscoSpark\Logs\media\mdi_5880_118_temp_index_v2.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Default\JumpListIconsMostVisited\3366ea62-89e6-4de5-84b6-9f3e1eb04751.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Default\JumpListIconsMostVisited\4444ea197-cb9e-4f7f-9eae-d705c9be2af9.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Default\JumpListIconsMostVisited\5b36e48f-554e-4f85-9630-ed0c7712e886.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Default\JumpListIconsMostVisited\ad312292-ee58-48cf-be5e-01346a29189e.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Default\JumpListIconsMostVisited\fe9f1469-98c3-4937-a5e5-023829a64d25.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Default\JumpListIconsRecentClosed\9b17c1a9-10d4-4685-8566-8376743359a5.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Default\JumpListIconsRecentClosed\d76329ac-9bb3-4f3a-8f00-d12080c95b7b.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Default\JumpListIconsRecentClosed\d904ee7b-4d90-433a-be60-fc66ad94eb11.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Default\Network\0b88e5e3-a142-4424-ab94-e788057c553a.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Default\Network\c45dbadd-ba58-4704-bc54-6d766e98dd78.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Profile 1\JumpListIconsRecentClosed\341c5f6d-3fca-4849-a2b8-c63a44836113.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Profile 1\JumpListIconsRecentClosed\3dca8358-c1c7-4cb9-b287-4f2da8f8722a.tmp
Deleted file - C:\Users\Student\AppData\Local\Google\Chrome\User Data\Profile 1\JumpListIconsRecentClosed\86a9e986-5b8d-b6dc-a7ba-3604f4c63e57.tmp
Deleted file - C:\Users\Student\AppData\Local\Microsoft\Edge\User Data\Default\JumpListIconsRecentClosed\4774a39b8-87e8-4db3-aac4-d8e44a30a69a.tmp
Deleted file - C:\Users\Student\AppData\Local\Microsoft\Edge\User Data\Default\JumpListIconsRecentClosed\ed8d95b9b-6b3d-4f83-ae0a-50420a82af28.tmp
Deleted file - C:\Users\Student\AppData\Local\Microsoft\Edge\User Data\Default\JumpListIconsTopSites\52c98359-15d6-4a90-a85c-0a2d5f80628a.tmp
Deleted file - C:\Users\Student\AppData\Local\Microsoft\Edge\User Data\Default\JumpListIconsTopSites\8d521016-ce0c-40b3-8705-571a560317c5.tmp
Deleted file - C:\Users\Student\AppData\Local\Microsoft\Edge\User Data\Default\JumpListIconsTopSites\b7ea109b-c237-4e02-a476-f79c4119807f.tmp
Deleted file - C:\Users\Student\AppData\Local\Microsoft\Edge\User Data\Default\Network\b7895a9c-9ccc-4469-a81e-9e30313a0942.tmp
Deleted file - C:\Users\Student\AppData\Local\Microsoft\Edge\User Data\Default\Network\f764f4fe-2f81-40e6-9ab4-ffef55c99fd2.tmp
.\Users\Student\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\11FE8C73.tmp
he process cannot access the file because it is being used by another process.
.\Users\Student\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\14B1D2A4.tmp
he process cannot access the file because it is being used by another process.
.\Users\Student\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\15DF1F6F.tmp
he process cannot access the file because it is being used by another process.
.\Users\Student\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\2BC0FFA6.tmp
he process cannot access the file because it is being used by another process.
.\Users\Student\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\38E9FA80.tmp
he process cannot access the file because it is being used by another process.
.\Users\Student\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\41CFC42B.tmp
he process cannot access the file because it is being used by another process.
.\Users\Student\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\49C98F37.tmp
he process cannot access the file because it is being used by another process.
.\Users\Student\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\4A7187F4.tmp
he process cannot access the file because it is being used by another process.
.\Users\Student\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\58E943BA.tmp
he process cannot access the file because it is being used by another process.
.\Users\Student\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\5EB925FF.tmp
he process cannot access the file because it is being used by another process.
.\Users\Student\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\69C571F9.tmp
he process cannot access the file because it is being used by another process.
.\Users\Student\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\6B496BBB.tmp

```

Fig. 11: `del /q /s *.tmp` – quietly deletes all temp files in subfolders [8].

Network Testing and Monitoring

Network diagnostics were performed using `ipconfig /all` to display comprehensive configuration details, `ping -n 10 google.com` to test connectivity and latency, and `netstat -ano` to view active connections with associated process IDs. These commands were instrumental in diagnosing intermittent Wi-Fi connectivity problems.

```
C:\Users\Student> ipconfig /all

Windows IP Configuration

    Host Name . . . . . : PC-PF5KK
    Primary Dns Suffix . . . . . :
    Node Type . . . . . : Hybrid
    IP Routing Enabled. . . . . : No
    WINS Proxy Enabled. . . . . : No

Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . :
    Description . . . . . : Realtek PCIe GbE Family Controller
    Physical Address. . . . . : C8-53-09-D7-3E-0E
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . :
    Description . . . . . : Microsoft Wi-Fi Direct Virtual Adapter
    Physical Address. . . . . : 4E-23-38-BB-FA-4D
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes

Wireless LAN adapter Local Area Connection* 2:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . :
    Description . . . . . : Microsoft Wi-Fi Direct Virtual Adapter #2
    Physical Address. . . . . : 4E-23-38-BB-EA-5D
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes
```

Fig. 12: `ipconfig /all` – shows full network details including DNS [10].

```
C:\Users\Student> ping -n 10 google.com

Pinging google.com [142.250.117.100] with 32 bytes of data:
Reply from 142.250.117.100: bytes=32 time=33ms TTL=109
Reply from 142.250.117.100: bytes=32 time=32ms TTL=109
Reply from 142.250.117.100: bytes=32 time=34ms TTL=109
Reply from 142.250.117.100: bytes=32 time=31ms TTL=109
Reply from 142.250.117.100: bytes=32 time=46ms TTL=109
Reply from 142.250.117.100: bytes=32 time=35ms TTL=109
Reply from 142.250.117.100: bytes=32 time=41ms TTL=109
Reply from 142.250.117.100: bytes=32 time=62ms TTL=109
Reply from 142.250.117.100: bytes=32 time=30ms TTL=109
Reply from 142.250.117.100: bytes=32 time=55ms TTL=109

Ping statistics for 142.250.117.100:
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 30ms, Maximum = 62ms, Average = 39ms

C:\Users\Student>
```

Fig. 13: ping -n 10 google.com – sends 10 pings to test connection.

```
C:\Users\Student> netstat -ano
```

Active Connections

Proto	Local Address	Foreign Address	State	PID
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING	1760
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING	4
TCP	0.0.0.0:5040	0.0.0.0:0	LISTENING	6608
TCP	0.0.0.0:7680	0.0.0.0:0	LISTENING	20780
TCP	0.0.0.0:49664	0.0.0.0:0	LISTENING	1488
TCP	0.0.0.0:49665	0.0.0.0:0	LISTENING	1312
TCP	0.0.0.0:49666	0.0.0.0:0	LISTENING	2572
TCP	0.0.0.0:49667	0.0.0.0:0	LISTENING	3144
TCP	0.0.0.0:49671	0.0.0.0:0	LISTENING	5364
TCP	0.0.0.0:49672	0.0.0.0:0	LISTENING	1460
TCP	127.0.0.1:1216	127.0.0.1:54065	ESTABLISHED	5084
TCP	127.0.0.1:7369	127.0.0.1:7370	ESTABLISHED	4584
TCP	127.0.0.1:7370	127.0.0.1:7369	ESTABLISHED	4584
TCP	127.0.0.1:15292	0.0.0.0:0	LISTENING	6856
TCP	127.0.0.1:15393	0.0.0.0:0	LISTENING	6856
TCP	127.0.0.1:16494	0.0.0.0:0	LISTENING	6856
TCP	127.0.0.1:19294	0.0.0.0:0	LISTENING	10340
TCP	127.0.0.1:27015	0.0.0.0:0	LISTENING	6792
TCP	127.0.0.1:45623	0.0.0.0:0	LISTENING	1860
TCP	127.0.0.1:52975	0.0.0.0:0	LISTENING	1860
TCP	127.0.0.1:54065	0.0.0.0:0	LISTENING	1860
TCP	127.0.0.1:54065	127.0.0.1:1216	ESTABLISHED	1860
TCP	127.0.0.1:54066	0.0.0.0:0	LISTENING	1860
TCP	127.0.0.1:57207	127.0.0.1:57208	ESTABLISHED	4584
TCP	127.0.0.1:57208	127.0.0.1:57207	ESTABLISHED	4584
TCP	127.0.0.1:57209	127.0.0.1:57210	ESTABLISHED	4584
TCP	127.0.0.1:57210	127.0.0.1:57209	ESTABLISHED	4584
TCP	127.0.0.1:57211	127.0.0.1:57212	ESTABLISHED	4584
TCP	127.0.0.1:57212	127.0.0.1:57211	ESTABLISHED	4584
TCP	127.0.0.1:57213	127.0.0.1:57214	ESTABLISHED	4584
TCP	127.0.0.1:57214	127.0.0.1:57213	ESTABLISHED	4584
TCP	127.0.0.1:57215	127.0.0.1:57216	ESTABLISHED	4584
TCP	127.0.0.1:57216	127.0.0.1:57215	ESTABLISHED	4584

Fig. 14: netstat -ano – shows active connections with process IDs.

User Management

The commands *net user* and *whoami /all* were used to enumerate local user accounts and display detailed information regarding the current user's privileges and group memberships.

```
C:\Users\Student> net user

User accounts for \\PC-PF5KK

-----
Administrator          DefaultAccount          Guest
Student                 WDAGUtilityAccount
The command completed successfully.

C:\Users\Student>
```

Fig. 15: *net user* – lists all local users

```
C:\Users\Student> whoami /all

USER INFORMATION
-----
User Name          SID
-----
pc-pf5kk\student  S-1-5-21-572943906-1319465739-1045318031-1001

GROUP INFORMATION
-----
Group Name          Type          SID          Attributes
-----
Everyone            Well-known group  S-1-1-0      Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Local account and member of Administrators group  Well-known group  S-1-5-114    Group used for deny only
BUILTIN\Administrators  Alias          S-1-5-32-544  Group used for deny only
BUILTIN\Users          Alias          S-1-5-32-545  Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\INTERACTIVE Well-known group  S-1-5-4      Mandatory group, Enabled by default, Enabled group
CONSOLE LOGON         Well-known group  S-1-2-1      Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users  Well-known group  S-1-5-11     Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization  Well-known group  S-1-5-15     Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Local account  Well-known group  S-1-5-113    Mandatory group, Enabled by default, Enabled group
LOCAL                Well-known group  S-1-2-0      Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NTLM Authentication  Well-known group  S-1-5-64-10  Mandatory group, Enabled by default, Enabled group
Mandatory Label\Medium Mandatory Level  Label          S-1-16-8192

PRIVILEGES INFORMATION
-----
Privilege Name      Description          State
-----
SeShutdownPrivilege  Shut down the system  Disabled
SeChangeNotifyPrivilege  Bypass traverse checking  Enabled
SeUndockPrivilege     Remove computer from docking station  Disabled
SeIncreaseWorkingSetPrivilege  Increase a process working set  Disabled
SeTimeZonePrivilege   Change the time zone  Disabled

C:\Users\Student>
```

Fig. 16 : *whoami /all* – shows current user and groups/privileges.[9].

Task Scheduling

Scheduled tasks were examined with `schtasks /query /fo LIST`, while the syntax for creating new tasks was tested using `schtasks /create`.

Overall, the use of appropriate switches (such as `/f` for force and `-n` for count) significantly enhanced the functionality of basic commands. The practical application of these tools demonstrated how the Windows CLI supports efficient system administration without reliance on graphical interfaces. All commands executed as expected when entered correctly, reinforcing the value of command-line proficiency for troubleshooting and maintenance tasks [9], [10].

```
Logon Mode: Interactive/Background
HostName: PC-PF5KK
TaskName: \Microsoft\Windows\WwanSvc\OobeDiscovery
Next Run Time: N/A
Status: Ready
Logon Mode: Interactive/Background

HostName: PC-PF5KK
TaskName: \Microsoft\Windows\WwanSvc\OobeDiscovery
Next Run Time: N/A
Status: Ready
Logon Mode: Interactive/Background

Folder: \Microsoft\XblGameSave
HostName: PC-PF5KK
TaskName: \Microsoft\XblGameSave\XblGameSaveTask
Next Run Time: N/A
Status: Ready
Logon Mode: Interactive/Background

Folder: \SoftLanding
INFO: There are no scheduled tasks presently available at your access level.

Folder: \SoftLanding\S-1-5-21-572943906-1319465739-1045318031-1001
HostName: PC-PF5KK
TaskName: \SoftLanding\S-1-5-21-572943906-1319465739-1045318031-1001\SoftLandingCreativeManagementTask
Next Run Time: 23/04/2026 15:12:00
Status: Ready
Logon Mode: Interactive only

HostName: PC-PF5KK
TaskName: \SoftLanding\S-1-5-21-572943906-1319465739-1045318031-1001\SoftLandingCreativeManagementTask
Next Run Time: 23/04/2026 15:12:00
Status: Ready
Logon Mode: Interactive only

HostName: PC-PF5KK
TaskName: \SoftLanding\S-1-5-21-572943906-1319465739-1045318031-1001\SoftLandingCreativeManagementTask
Next Run Time: 23/04/2026 15:12:00
Status: Ready
Logon Mode: Interactive only

HostName: PC-PF5KK
TaskName: \SoftLanding\S-1-5-21-572943906-1319465739-1045318031-1001\SoftLandingDeferralTask-{706287a7-e6b5-4015-b314-c0d12852a50b}
Next Run Time: 23/04/2026 05:15:00
Status: Ready
Logon Mode: Interactive only

HostName: PC-PF5KK
TaskName: \SoftLanding\S-1-5-21-572943906-1319465739-1045318031-1001\SoftLandingTriggerTask-128000000001615609-render-{d959b7f6-c3d3-4ca7-93c7-1c3aa3bc6dff}
Next Run Time: N/A
Status: Disabled
Logon Mode: Interactive only

C:\Users\Student\
```

Fig. 17: `schtasks /query /fo LIST` – lists all scheduled tasks.

```

C:\Users\Student> schtasks /query /fo LIST

Folder: \
HostName:      PC-PF5KK
TaskName:      \Adobe Acrobat Update Task
Next Run Time: 23/04/2026 10:00:00
Status:        Ready
Logon Mode:    Interactive/Background

HostName:      PC-PF5KK
TaskName:      \Adobe Acrobat Update Task
Next Run Time: 23/04/2026 10:00:00
Status:        Ready
Logon Mode:    Interactive/Background

HostName:      PC-PF5KK
TaskName:      \Launch Adobe CCXProcess
Next Run Time: 24/04/2026 02:14:00
Status:        Ready
Logon Mode:    Interactive only

HostName:      PC-PF5KK
TaskName:      \OneDrive Per-Machine Standalone Update Task
Next Run Time: 23/04/2026 14:17:50
Status:        Ready
Logon Mode:    Interactive/Background

HostName:      PC-PF5KK
TaskName:      \OneDrive Reporting Task-S-1-5-21-572943906-1319465739-1045318031-1001
Next Run Time: 23/04/2026 15:44:25
Status:        Ready
Logon Mode:    Interactive only

HostName:      PC-PF5KK
TaskName:      \OneDrive Startup Task-S-1-5-21-572943906-1319465739-1045318031-1001
Next Run Time: N/A
Status:        Ready
Logon Mode:    Interactive only

Folder: \GoogleSystem
INFO: There are no scheduled tasks presently available at your access level.

Folder: \Lenovo
INFO: There are no scheduled tasks presently available at your access level.

Folder: \Lenovo\Vantage
INFO: There are no scheduled tasks presently available at your access level.

Folder: \Microsoft
INFO: There are 1 scheduled tasks presently available at your access level.

```

Fig. 18: `schtasks /query /fo LIST` – lists all scheduled tasks.

```

C:\Users\Student> schtasks /create /tn "MyTask" /tr "notepad.exe" /sc once /st 14:00
SUCCESS: The scheduled task "MyTask" has successfully been created.

C:\Users\Student>

```

Fig. 19: `schtasks /create /tn "MyTask" /tr "notepad.exe" /sc once /st 14:00` – creates a one-time task [8].

3. Use of Linux OS CLI Commands

A series of Bash commands were executed within an Ubuntu virtual machine to explore Linux command-line capabilities. A few commands with appropriate options and switches were tested across the required areas, with screenshots documenting both input and output [6], [11].

Process management

Process management in Linux was explored using several powerful commands. The ps aux command displays a comprehensive list of all running processes, including user ownership, CPU and memory utilisation. For real-time monitoring, the top command was employed, providing a dynamic view of system resources. Additionally, the kill -9 command was tested to demonstrate forceful termination of processes when necessary. These tools allow administrators to effectively monitor system performance and manage processes with precision [6], [11].

```
sysadmin@localhost:~$ ps aux
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root             1  0.0  0.0  17984  3012 pts/0    Ss   09:12   0:00 /sbin?? /ini
syslog        33  0.0  0.0  182132  2940 ?        Ssl  09:12   0:00 /usr/sbin/rsy
root         37  0.0  0.0   23676  2112 ?        Ss   09:12   0:00 /usr/sbin/cro
root         39  0.0  0.0   61404  3084 ?        Ss   09:12   0:00 /usr/sbin/ssh
bind         56  0.2  0.0  827548  40076 ?        Ssl  09:12   0:00 /usr/sbin/nam
root         77  0.0  0.0   63156  2916 pts/0    S    09:12   0:00 /bin/login -f
sysadmin     87  0.0  0.0   18200  3376 pts/0    S    09:12   0:00 -bash
sysadmin     98  0.0  0.0   15588  2152 pts/0    R+   09:13   0:00 ps aux
sysadmin@localhost:~$
```

Fig. 20: ps aux – shows all processes with user, CPU, memory [11].

```
Tasks:  8 total,  1 running,  7 sleeping,  0 stopped,  0 zombie
%Cpu(s):  0.9 us,  0.8 sy,  0.0 ni, 98.3 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem: 65851452 total, 7699208 used, 58152244 free, 216924 buffers
KiB Swap:  0 total,  0 used,  0 free. 2450644 cached Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM     TIME+ COMMAND
    1 root        20   0   17984   3012   2760 S   0.0   0.0   0:00.02 init
   33 syslog     20   0  182132   2940  2432 S   0.0   0.0   0:00.00 rsyslogd
   37 root       20   0   23676   2112  1892 S   0.0   0.0   0:00.00 cron
   39 root       20   0   61404   3084  2408 S   0.0   0.0   0:00.00 sshd
   56 bind        20   0  827548  40076  5268 S   0.0   0.1   0:00.08 named
   77 root       20   0   63156   2916  2460 S   0.0   0.0   0:00.00 login
   87 sysadmin   20   0   18200   3380  2888 S   0.0   0.0   0:00.00 bash
  100 sysadmin   20   0   19876   2400  2072 R   0.0   0.0   0:00.01 top
```

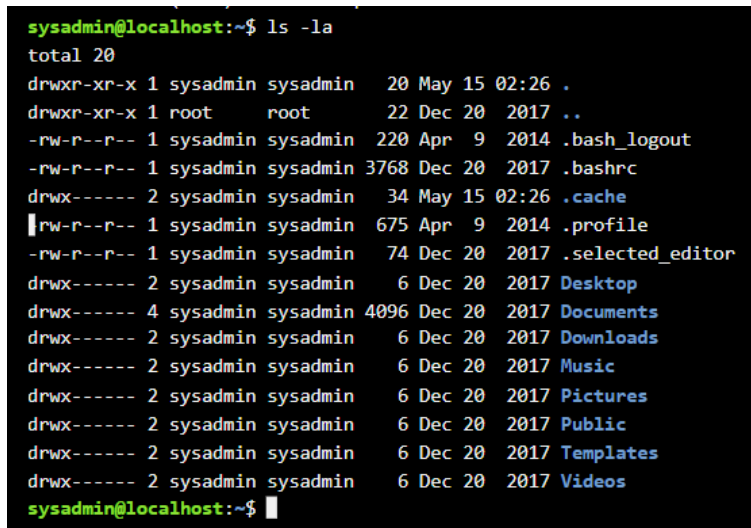
Fig. 21: top -- live view.

```
sysadmin@localhost:~$ kill -9
kill: usage: kill [-s sigspec | -n signum | -sigspec] pid | jobspec ... or kill
-l [sigspec]
sysadmin@localhost:~$
```

Fig. 22: kill -9 – force kill a process

File & directory management

File and directory operations form a fundamental part of Linux system administration. The `ls -la` command was used to list all files and directories with detailed permissions and ownership information. Recursive directory copying was performed using `cp -r`, while `mv` enabled renaming and moving of files and folders. The `mkdir -p` command facilitated the creation of nested directory structures, and `rm -rf` was demonstrated (with caution) for removing directories and their contents. These commands provide fine-grained control over the Linux file system [6].



```
sysadmin@localhost:~$ ls -la
total 20
drwxr-xr-x 1 sysadmin sysadmin 20 May 15 02:26 .
drwxr-xr-x 1 root root 22 Dec 20 2017 ..
-rw-r--r-- 1 sysadmin sysadmin 220 Apr 9 2014 .bash_logout
-rw-r--r-- 1 sysadmin sysadmin 3768 Dec 20 2017 .bashrc
drwx----- 2 sysadmin sysadmin 34 May 15 02:26 .cache
-rw-r--r-- 1 sysadmin sysadmin 675 Apr 9 2014 .profile
-rw-r--r-- 1 sysadmin sysadmin 74 Dec 20 2017 .selected_editor
drwx----- 2 sysadmin sysadmin 6 Dec 20 2017 Desktop
drwx----- 4 sysadmin sysadmin 4096 Dec 20 2017 Documents
drwx----- 2 sysadmin sysadmin 6 Dec 20 2017 Downloads
drwx----- 2 sysadmin sysadmin 6 Dec 20 2017 Music
drwx----- 2 sysadmin sysadmin 6 Dec 20 2017 Pictures
drwx----- 2 sysadmin sysadmin 6 Dec 20 2017 Public
drwx----- 2 sysadmin sysadmin 6 Dec 20 2017 Templates
drwx----- 2 sysadmin sysadmin 6 Dec 20 2017 Videos
sysadmin@localhost:~$
```

Fig. 23: `ls -la` – list all files including hidden with details [6].

Network testing/monitoring

Network-related activities were examined using several diagnostic commands. The `ip addr show` command displayed detailed information about network interfaces and IP addresses. Connectivity testing was conducted with `ping -c 5 google.com`, while `ss -tuln` was used to view listening sockets and active network services. These commands enable efficient troubleshooting of network issues and monitoring of system connectivity without graphical tools [10].

Linux command-line utilities offer significant flexibility and power. Commands such as `ps aux` and `top` deliver extensive real-time system information, while file and network tools allow rapid and precise administration. However, caution is required with potentially destructive commands such as `rm -rf`, which do not request confirmation before execution [6], [11].

```

sysadmin@localhost:~$ ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
20860: eth0@if20861: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 02:42:c0:a8:01:02 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.2/24 brd 192.168.1.255 scope global eth0
        valid_lft forever preferred_lft forever
sysadmin@localhost:~$

```

Fig. 24: `ip addr show` – shows network interfaces and IPs.

```

sysadmin@localhost:~$ ss -tln
Netid State      Recv-Q Send-Q   Local Address:Port   Peer Address:Port
tcp    UNCONN     0      0      192.168.1.2:53       *:*
tcp    UNCONN     0      0      127.0.0.1:53         *:*
tcp    UNCONN     0      0      127.0.0.11:38883     *:*
tcp    UNCONN     0      0      :::53                :::*
tcp    LISTEN     0      128     127.0.0.1:953        *:*
tcp    LISTEN     0      10      127.0.0.1:53         *:*
tcp    LISTEN     0  4096     127.0.0.11:37197     *:*
tcp    LISTEN     0      10      192.168.1.2:53       *:*
tcp    LISTEN     0      128     *:22                  *:*
tcp    LISTEN     0      128     :::953                :::*
tcp    LISTEN     0      128     :::22                  :::*
tcp    LISTEN     0      10      :::53                  :::*
sysadmin@localhost:~$

```

Fig. 25: `ss -tln` – shows listening sockets (replaces `netstat`) [10].

User management

```

tcp    LISTEN     0      10
sysadmin@localhost:~$ whoami
sysadmin
sysadmin@localhost:~$

```

Fig. 26: `whoami` – current user.

```

sysadmin
sysadmin@localhost:~$ id
uid=1001(sysadmin) gid=1001(sysadmin) groups=1001(sysadmin),4(adm),27(sudo)
sysadmin@localhost:~$

```

Fig. 27: `id` – user ID and groups.

Task scheduling

```
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
~
"/tmp/crontab.W8KwV/crontab" 22L, 888C
```

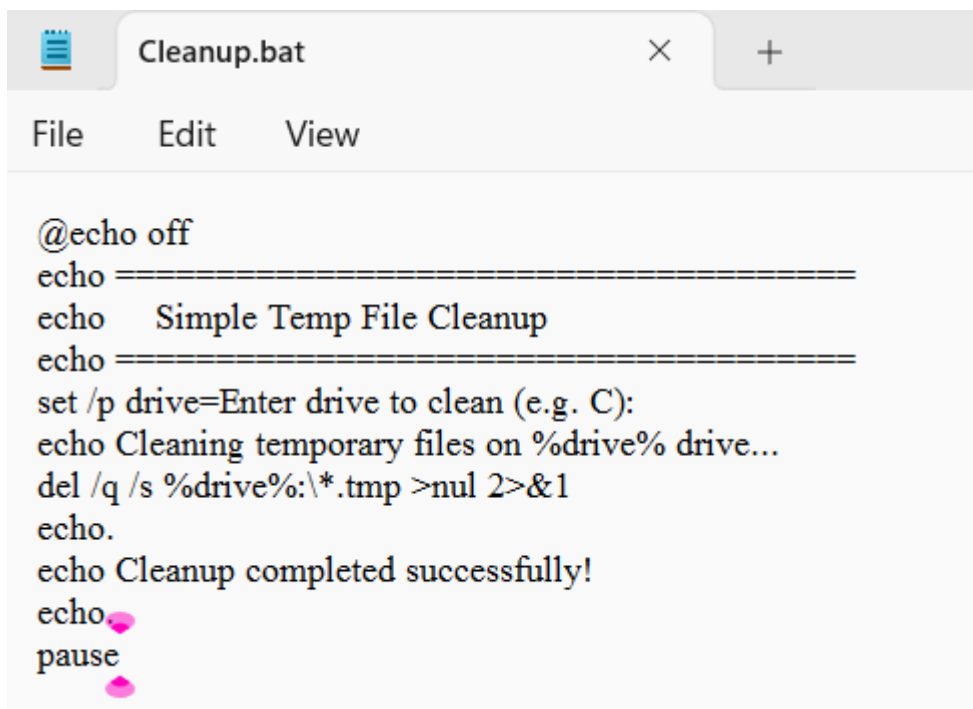
Fig. 28: `crontab -e` – edit to add jobs (I added a test one) [11].

4. Use of Windows OS Scripts

Two Batch scripts were developed to demonstrate the practical application of Windows command-line automation. Both scripts incorporate user input, variables, conditional logic or loops, and were designed to perform useful system maintenance tasks.

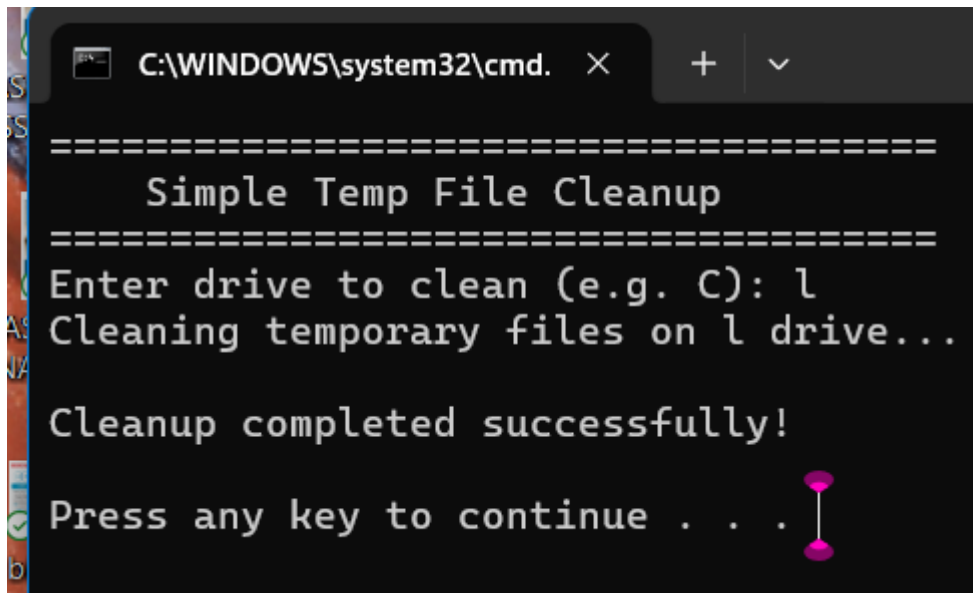
The first script automates the cleanup of temporary files. It prompts the user to specify a drive letter, recursively deletes files with the .tmp extension using the del command with appropriate switches, and provides confirmation of the operation. This script illustrates how Batch programming can efficiently handle routine housekeeping tasks that would otherwise require manual intervention.

The second script facilitates the backup of files from a source directory to a designated backup location. It requests both source and destination paths from the user, creates the destination folder if it does not exist, and utilises a for loop to copy all files while providing real-time feedback. Both scripts were written in Notepad, saved with the .bat extension, and executed successfully in Command Prompt. Screenshots were captured showing the source code and the resulting output. These examples highlight the effectiveness of Batch scripting for automating repetitive administrative procedures in the Windows environment.



```
@echo off
echo =====
echo   Simple Temp File Cleanup
echo =====
set /p drive=Enter drive to clean (e.g. C):
echo Cleaning temporary files on %drive% drive...
del /q /s %drive%:\*.tmp >nul 2>&1
echo.
echo Cleanup completed successfully!
echo.
pause
```

Fig. 29: Windows Batch Script code for cleanup drive



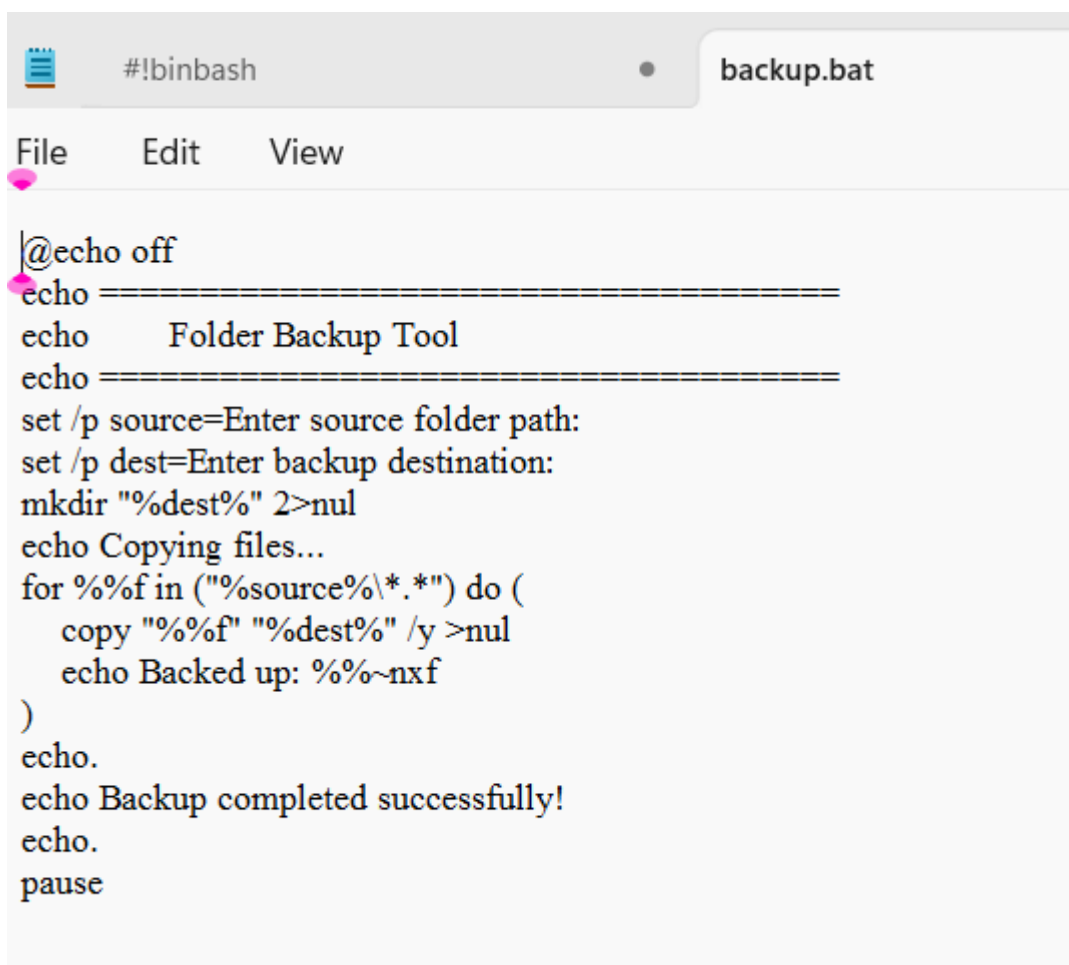
A screenshot of a Windows Command Prompt window. The title bar shows the path 'C:\WINDOWS\system32\cmd.' and standard window controls. The command prompt displays a batch script titled 'Simple Temp File Cleanup'. The script prompts the user to enter a drive letter (e.g., C), which is 'l' in this case. It then reports that cleanup is completed successfully and prompts the user to press any key to continue. A pink cursor is visible at the end of the 'Press any key to continue' line.

```
=====
Simple Temp File Cleanup
=====
Enter drive to clean (e.g. C): l
Cleaning temporary files on l drive...

Cleanup completed successfully!

Press any key to continue . . .
```

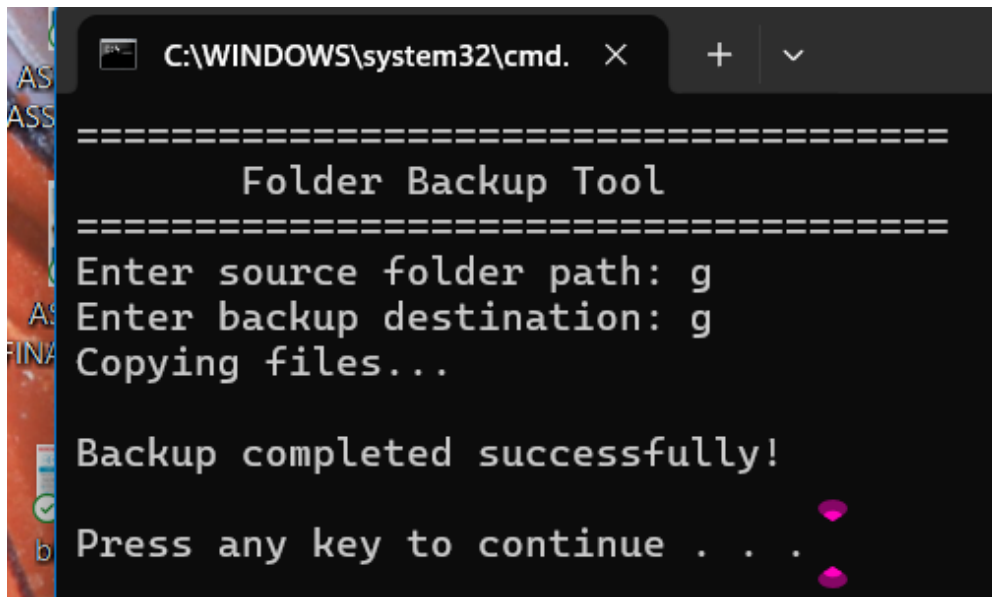
Fig. 30: Windows code execution



A screenshot of a Windows Batch Script editor window. The title bar shows the file name 'backup.bat' and the shell '#!binbash'. The menu bar includes 'File', 'Edit', and 'View'. The script content is as follows:

```
@echo off
echo =====
echo      Folder Backup Tool
echo =====
set /p source=Enter source folder path:
set /p dest=Enter backup destination:
mkdir "%dest%" 2>nul
echo Copying files...
for %%f in ("%source%\*.*") do (
    copy "%%f" "%dest%" /y >nul
    echo Backed up: %%~nxf
)
echo.
echo Backup completed successfully!
echo.
pause
```

Fig. 31: Windows Batch Script for backup drive



```
C:\WINDOWS\system32\cmd. X + v
=====
Folder Backup Tool
=====
Enter source folder path: g
Enter backup destination: g
Copying files...

Backup completed successfully!

Press any key to continue . . .
```

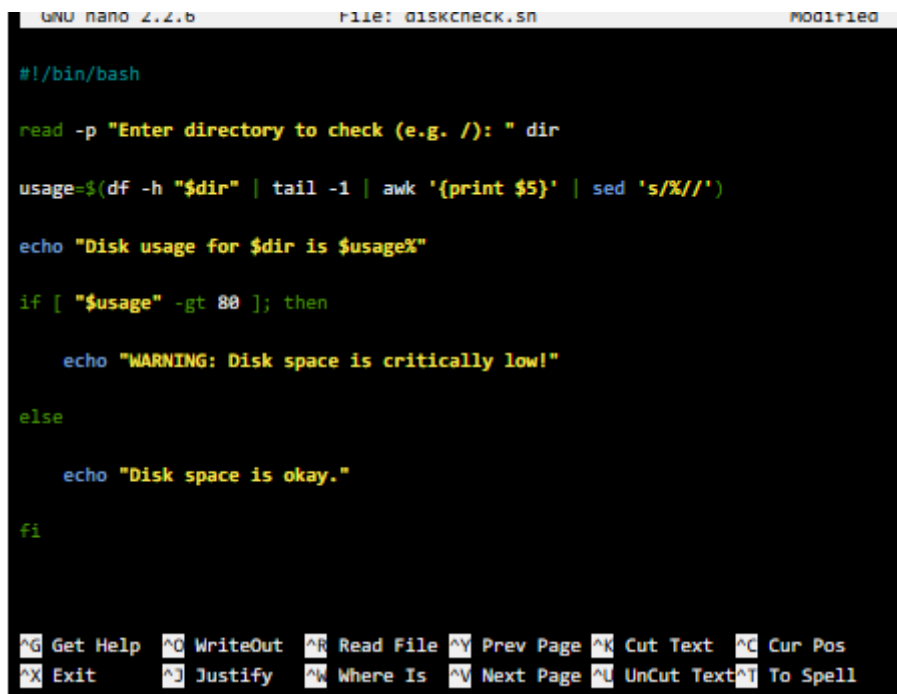
Fig. 32: Windows code Script execution for buckup drive

5. Use of Linux OS Scripts

Two Bash scripts were created to demonstrate automation and control flow within the Linux terminal environment. Both scripts were developed using the nano editor, made executable, and tested in the Ubuntu virtual machine.

The first script evaluates disk usage for a user-specified directory. It prompts for input, calculates the usage percentage using the `df` command combined with text processing tools, and issues a warning if usage exceeds 80%. This provides a simple yet practical monitoring tool for system administrators.

The second script performs a basic file backup operation. It accepts source and destination paths from the user, ensures the backup directory exists, and employs a `for` loop to copy files while displaying progress. Screenshots were taken showing the script code in the nano editor and the corresponding execution output in the terminal. These scripts demonstrate the flexibility and power of Bash scripting for automating common tasks in Linux environments [11].



```
GNU nano 2.2.6      file: diskcheck.sh      Modified

#!/bin/bash

read -p "Enter directory to check (e.g. /): " dir

usage=$(df -h "$dir" | tail -1 | awk '{print $5}' | sed 's/%//')

echo "Disk usage for $dir is $usage%"

if [ "$usage" -gt 80 ]; then

    echo "WARNING: Disk space is critically low!"

else

    echo "Disk space is okay."

fi

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^N Next Page  ^L UnCut Text ^T To Spell
```

Fig. 33:: Linux OS Bash nano code script

```
sysadmin@localhost:~$ read -p "Enter source folder: " src
Enter source folder: sysadmin@localhost:~$
sysadmin@localhost:~$ read -p "Enter backup folder: " bak
Enter backup folder: sysadmin@localhost:~$
sysadmin@localhost:~$ mkdir -p "$bak"
sysadmin@localhost:~$
sysadmin@localhost:~$ for file in "$src"/*; do
>
>   if [ -f "$file" ]; then
>
>       cp "$file" "$bak/"
>
>       echo "Backed up: $(basename "$file")"
>
>   fi
>
> done
sysadmin@localhost:~$
sysadmin@localhost:~$ echo "Backup finished successfully!"
Backup finished successfully!
sysadmin@localhost:~$
```

Fig. 34: Linux OS code execution output

Conclusion

This report has examined the fundamental functions of general-purpose operating systems and demonstrated practical command-line and scripting techniques in both Windows and Linux environments. The investigation revealed that despite differences in user interfaces, modern operating systems perform remarkably similar core tasks, including process management, memory management, file system control, device handling, networking, and security.

The command-line exercises highlighted the efficiency and precision offered by CLI tools. Windows commands such as `tasklist`, `ipconfig`, and `schtasks`, alongside Linux equivalents like `ps aux`, `ip addr show`, and `crontab`, allow rapid system administration without reliance on graphical interfaces. Furthermore, the development of Batch and Bash scripts illustrated how automation can transform repetitive tasks into efficient, repeatable processes. These scripts, incorporating user input, variables, loops, and conditional logic, demonstrated clear practical value for system maintenance and backup operations.

Overall, the hands-on activities significantly enhanced understanding of operating system architecture and strengthened technical skills in using the command line. The experience confirmed that proficiency in CLI tools and scripting is essential for effective IT administration. Engaging with multiple sources and completing the Cisco Linux Unhatched course provided a well-rounded perspective on both theoretical concepts and real-world application. Moving forward, these competencies will serve as a strong foundation for more advanced studies in computing and system administration.

Reference List

- [1] Microsoft, “Windows operating system overview,” Microsoft Learn, 2025. [Online]. Available: <https://learn.microsoft.com/en-us/windows/win32/sysinfo/windows-overview>
- [2] A. Silberschatz, P. B. Galvin, and G. Gagne, *Operating System Concepts*, 10th ed. Hoboken, NJ, USA: Wiley, 2018.
- [3] GeeksforGeeks, “Functions of Operating System,” Aug. 2025. [Online]. Available: <https://www.geeksforgeeks.org/functions-of-operating-system/>
- [4] DigitalOcean, “Understanding Process Management in Operating Systems,” 2024. [Online]. Available: <https://www.digitalocean.com/community/tutorials/process-management>
- [5] TutorialsPoint, “Memory Management in Operating Systems,” 2025. [Online]. Available: https://www.tutorialspoint.com/operating_system/os_memory_management.htm
- [6] Linux Foundation, “Linux File System and Directory Management,” 2025. [Online]. Available: <https://www.linuxfoundation.org/>
- [7] Britannica, “Operating System Functions,” 2024. [Online]. Available: <https://www.britannica.com/technology/operating-system>
- [8] Microsoft, “Windows commands reference,” Microsoft Learn, Apr. 2026. [Online]. Available: <https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/>
- [9] SS64, “PowerShell and CMD commands for system administration,” 2025. [Online]. Available: <https://ss64.com/>
- [10] NetworkChuck, “Network troubleshooting commands in Windows and Linux,” 2025. [Online]. Available: <https://networkchuck.com/>
- [11] Baeldung, “Bash commands cheat sheet,” Mar. 2026. [Online]. Available: <https://www.baeldung.com/linux/bash-commands>

Appendices

Appendix 1: Tutorial Activity – OS Types from OS Tutorial

Appendix 2: Lecture Activities / Activity Log

Appendix 3: Evidence of Engagement with Online Certified Course

The Cisco Networking Academy “Linux Unhatched” course was almost completed as part of this module also engaged with the Linux essentials course that is self-pace. This course provided a foundational introduction to Linux commands, file system navigation, and basic system administration — directly supporting the practical elements of this report.

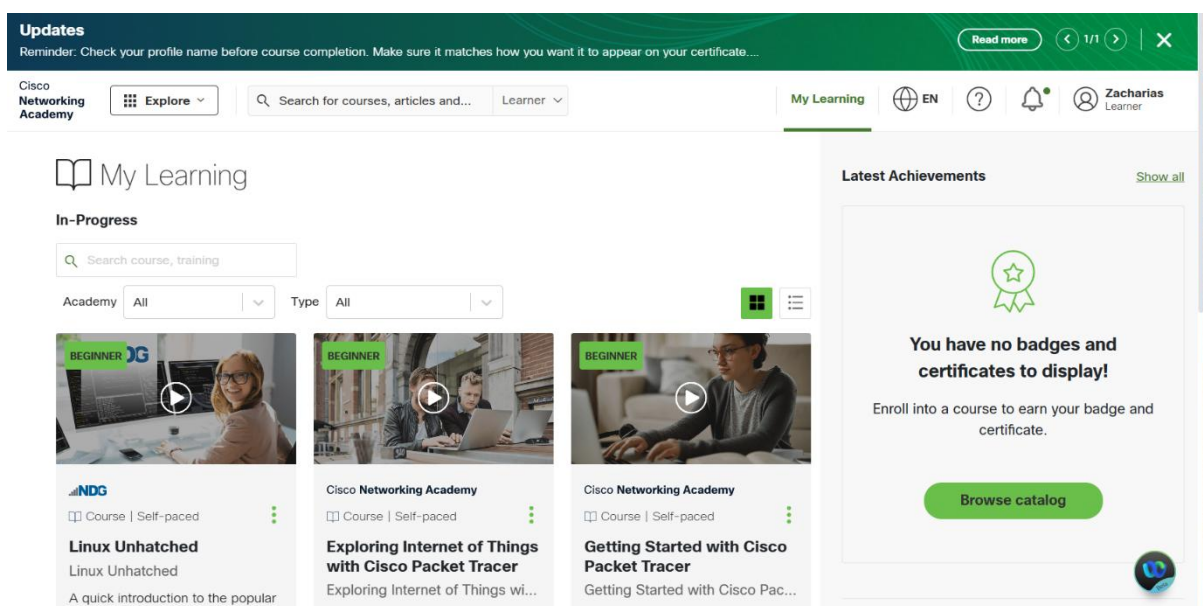


Fig. 35: Cisco account dashboard and selected modules

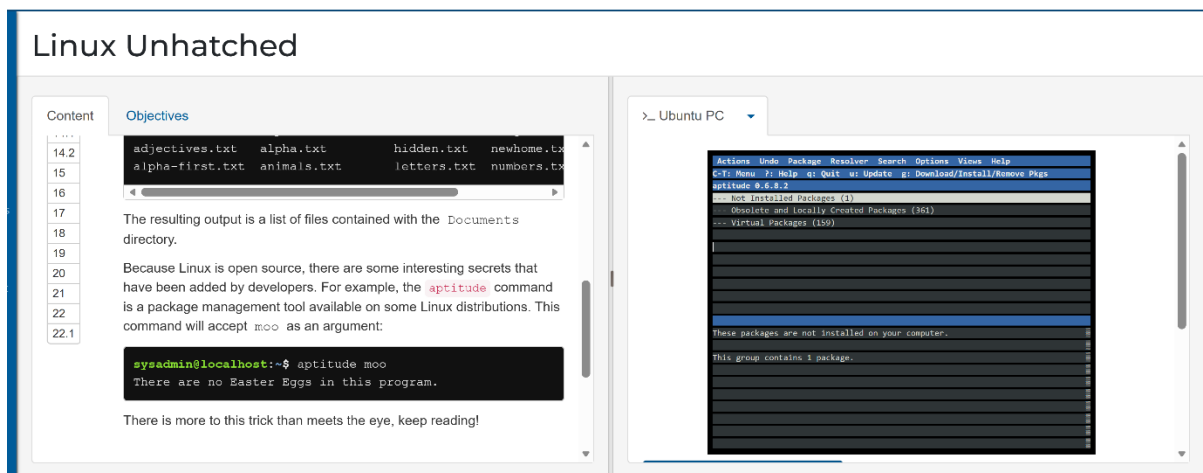


Fig. 36: Linux teaching progress

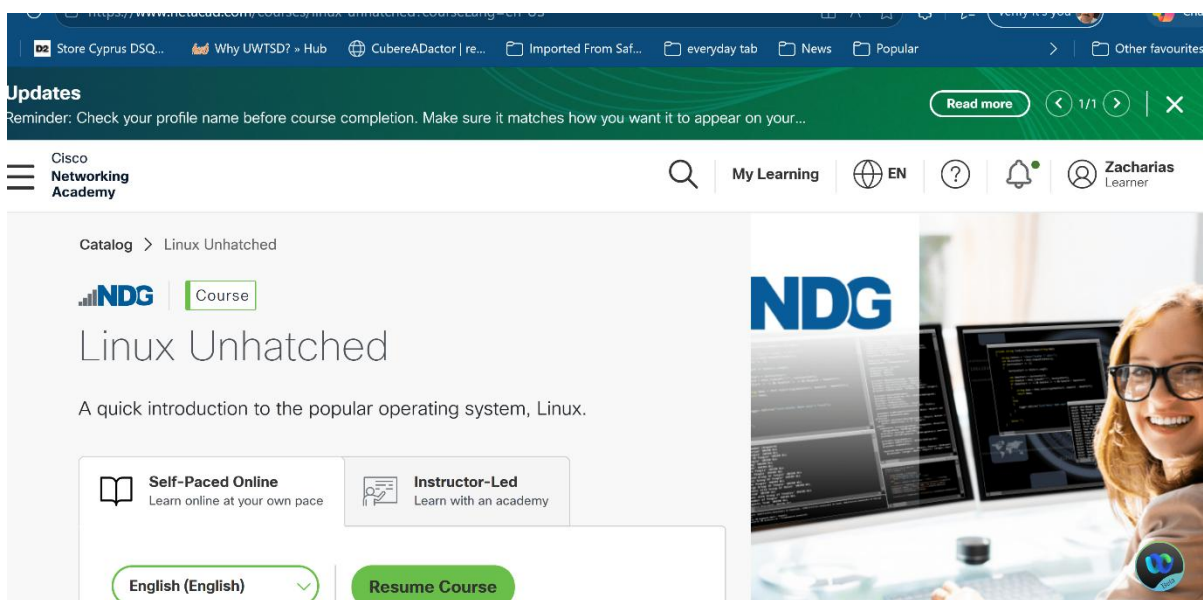


Fig. 37: Linux unhatched course

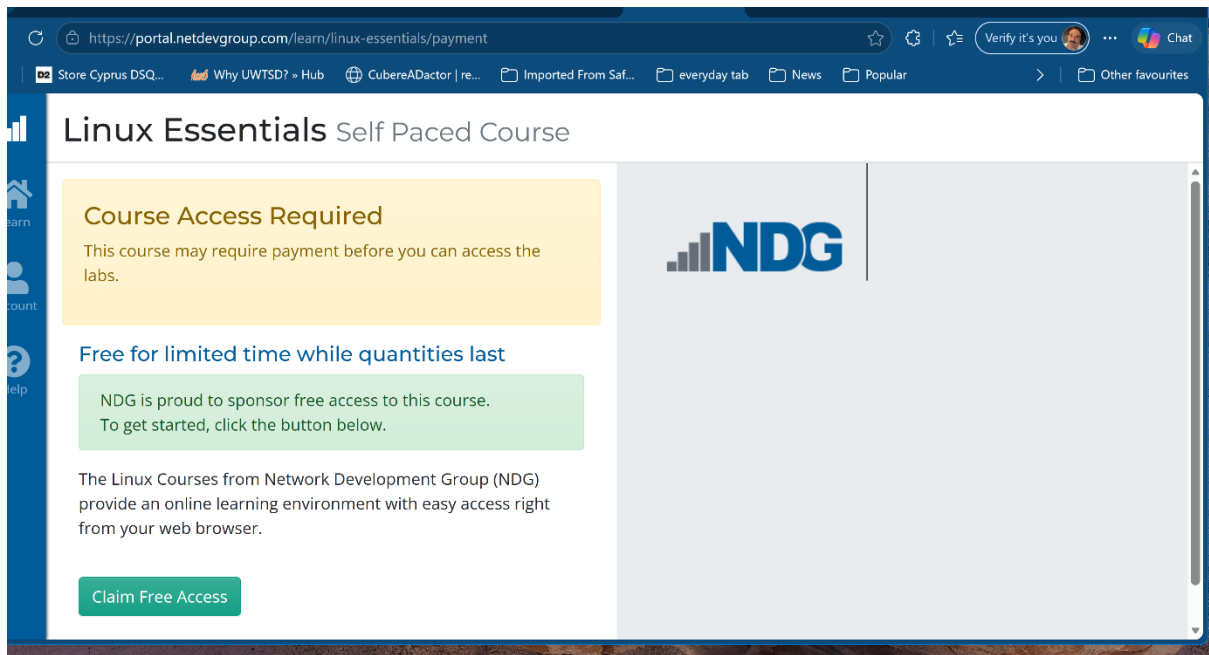


Fig. 38Linux Essentials course