Week 4 – Windows Server Basics (Simulated using PowerShell on ARM Laptop)
Omprakash Nara

Objective:

The primary goal of Week 4 was to gain hands-on experience with Windows Server features such as Active Directory Domain Services (AD DS), user and group management, and security policies like password rules. However, due to hardware limitations on my ARM-based laptop (Snapdragon processor), it is not possible to run virtualization software like VirtualBox or VMware Workstation. Therefore, I simulated all key tasks using Windows PowerShell, which allowed me to complete the same user/group management functions and password policy configurations locally.

This simulation helped me understand how system administrators perform user account creation, group assignments, and enforce basic security policies — which are common tasks in managing enterprise environments.

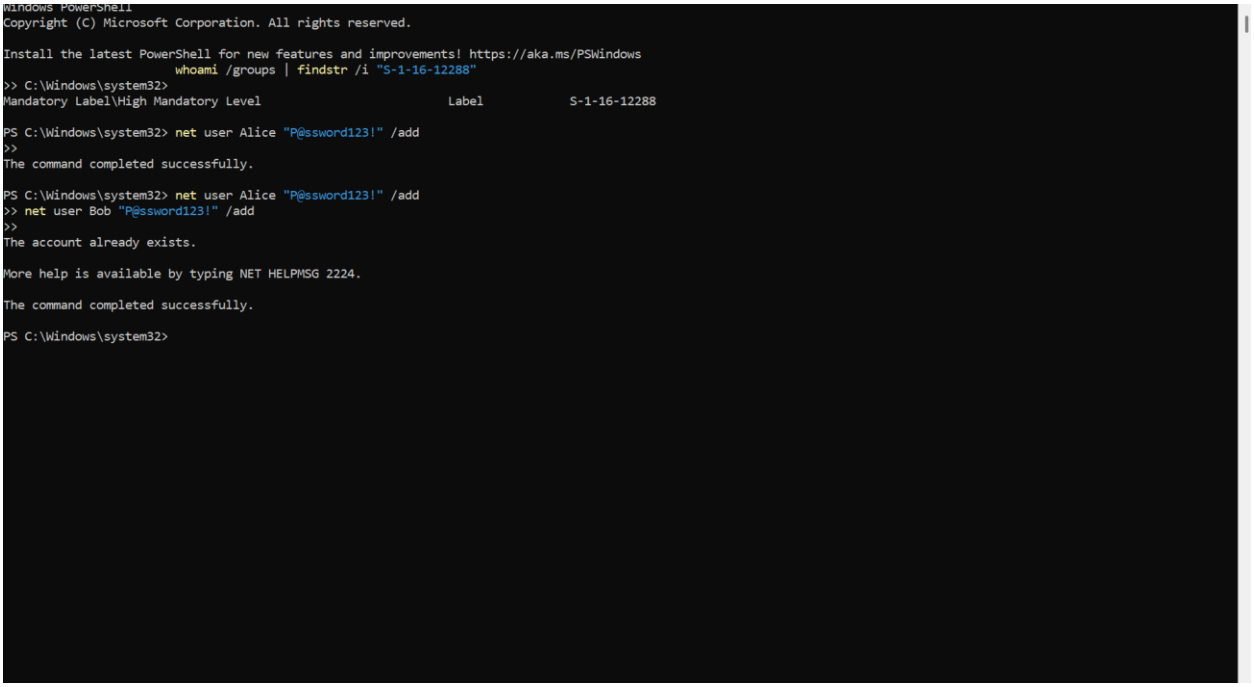
Tasks Completed:

1. Creating Local Users:

Using PowerShell, I created two new local user accounts named “Alice” and “Bob.” These users were configured with passwords that meet typical enterprise standards (minimum 8 characters, using a mix of characters and numbers).

The exact command used was:
net user Alice "P@ssword123!" /add
net user Bob "P@ssword123!" /add

This step simulates the kind of user provisioning an admin would do inside a domain environment or workgroup.



---

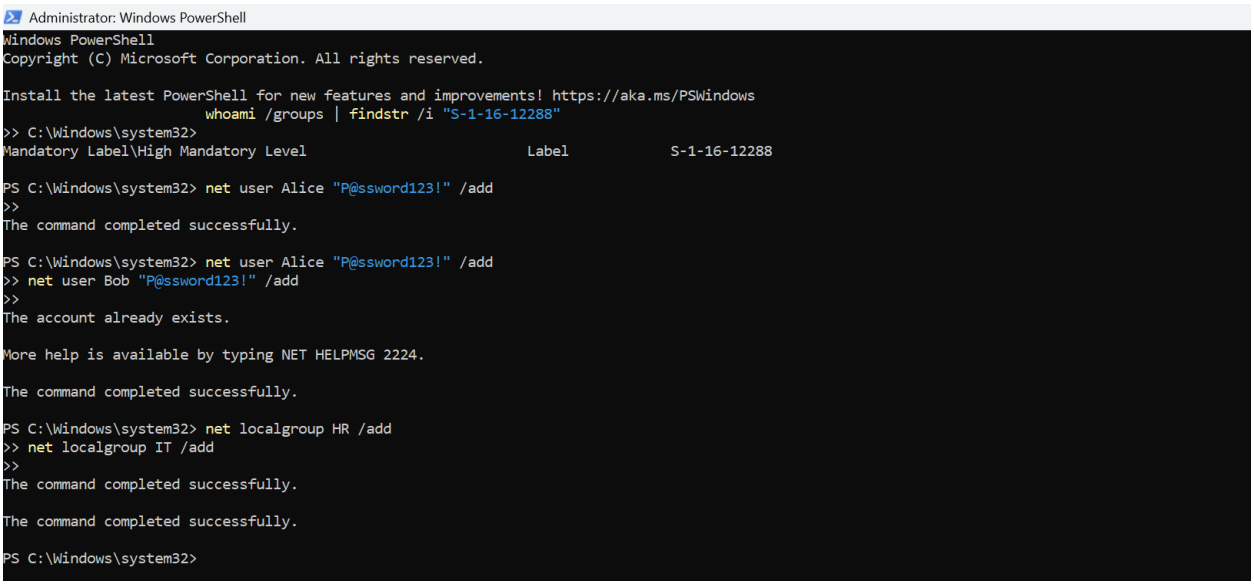
2. Creating Local Groups:

Next, I created two groups called “HR” and “IT” to represent organizational departments. These groups help control permissions and organize user access in real-world enterprise setups.

Groups created:

- HR
- IT

Command used:
net localgroup HR /add
net localgroup IT /add



---

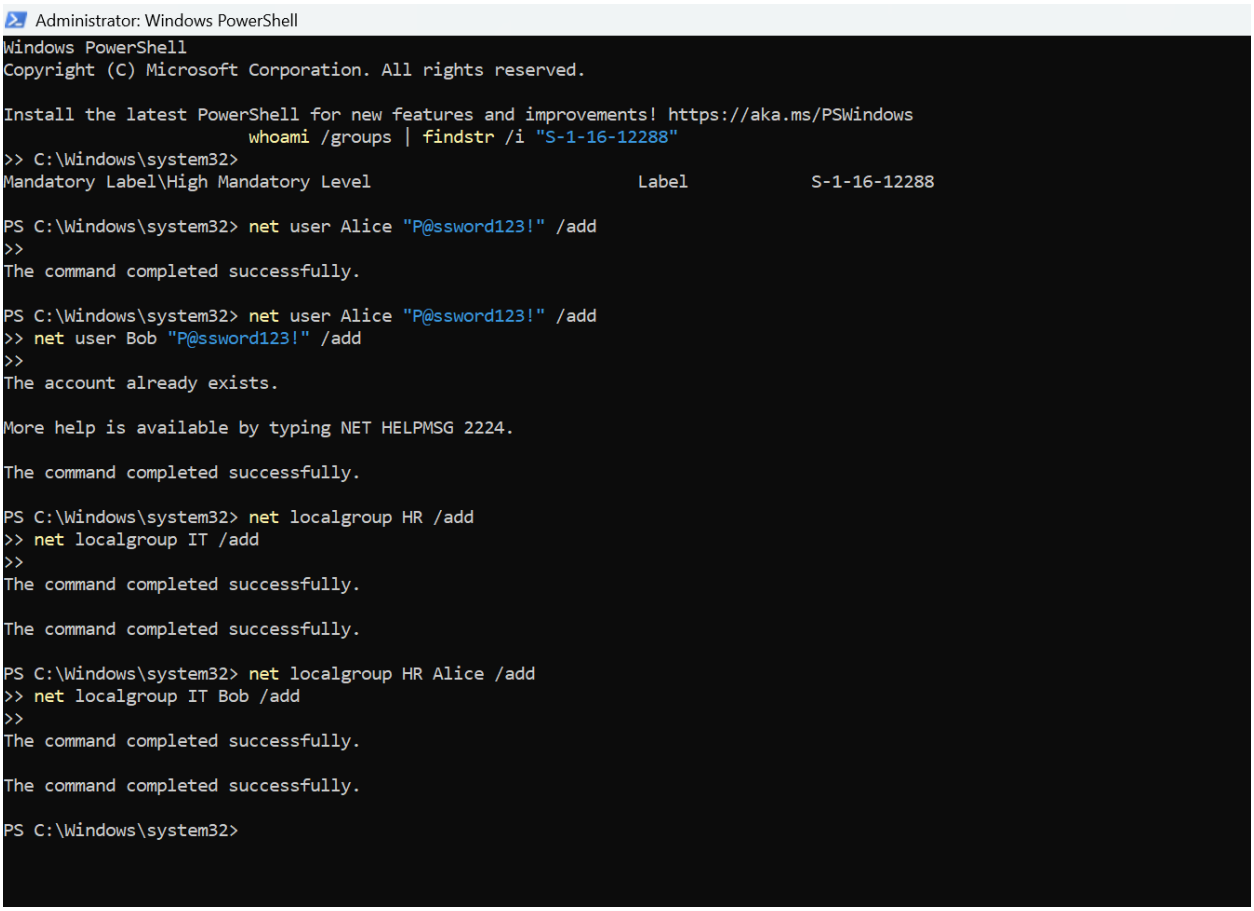
3. Assigning Users to Groups:

After creating both the users and groups, I assigned each user to their appropriate department. This helps simulate real-world identity and access management:

- Alice was added to the HR group
- Bob was added to the IT group

Command used:
net localgroup HR Alice /add
net localgroup IT Bob /add

This reflects how system engineers group users for role-based access control in Active Directory or local group policy systems.



---

4. Exporting and Reviewing Password Policy Settings:

Since I couldn't access Group Policy Management or Active Directory tools, I simulated password policy enforcement by exporting the local system's password and account settings using the secedit command.

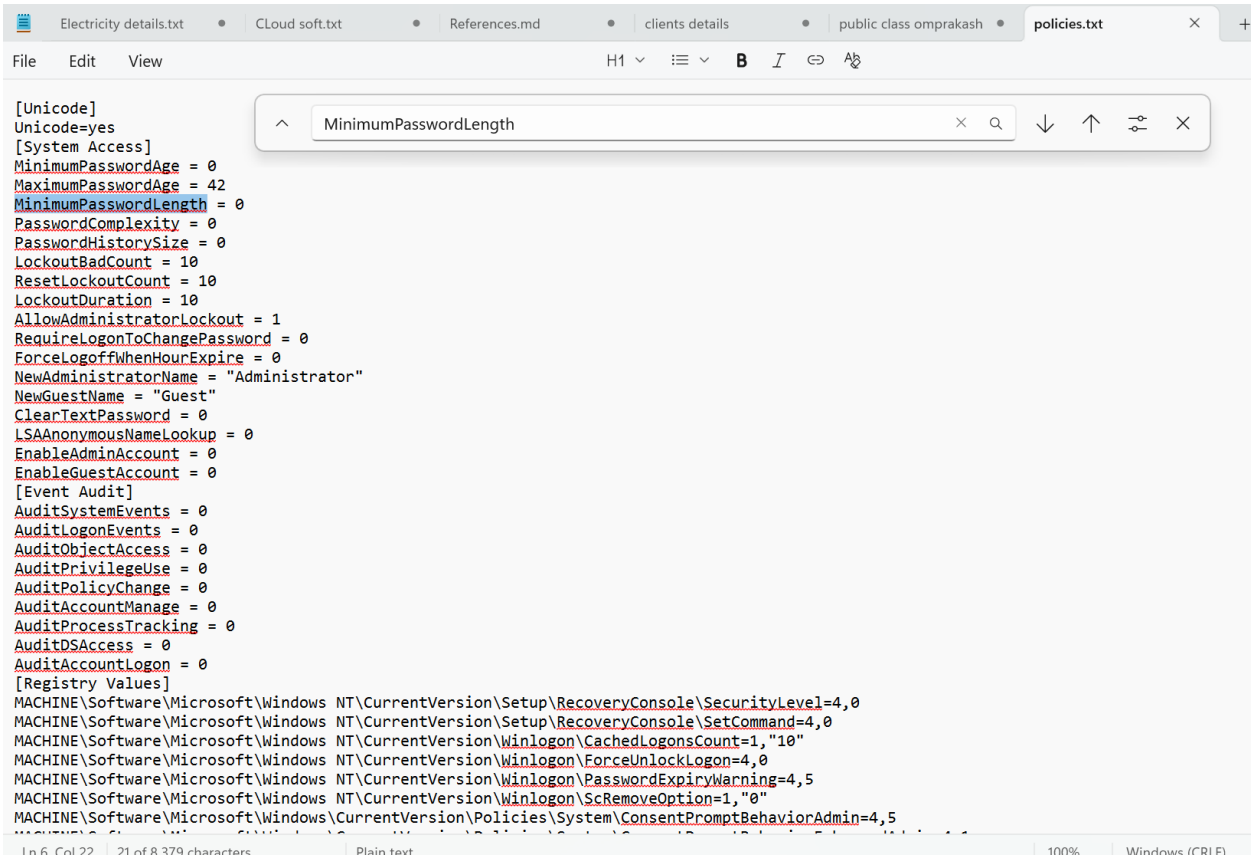
I ran:
secedit /export /cfg "C:\Users\ompra\OneDrive\Desktop\policies.txt" /log "C:\Users\ompra\OneDrive\Desktop\export_log.txt"

Then opened the exported file using Notepad to locate:

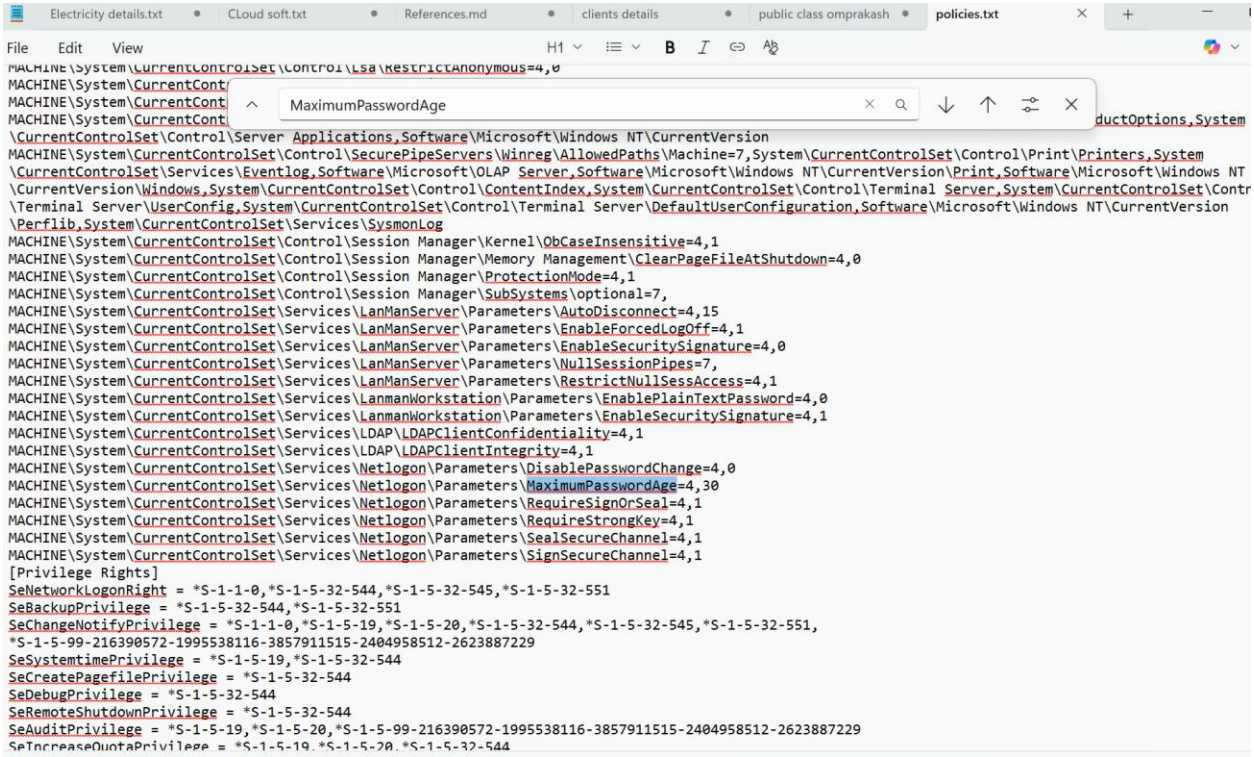
- MinimumPasswordLength
- MaximumPasswordAge

These settings help enforce good security practices across user accounts, ensuring users create strong passwords and change them regularly.

MinimumPasswordLength:



MaximumPasswordAge:



Outcome and Learnings:

Despite hardware limitations, I was able to successfully simulate the key concepts behind Windows Server administration using PowerShell. This included:

- Creating and managing user accounts
- Setting up and assigning users to local groups
- Reviewing system-wide password policies
- Understanding how user and group management is a critical part of identity and access control in enterprise networks

This exercise gave me confidence in using PowerShell for system administration and reinforced my understanding of how Windows Server environments are structured and maintained — even without a GUI or domain controller.