**Project Report: Active Directory Home lab**

**1. Executive Summary**

This project involved building a virtualised enterprise network environment from scratch to simulate a real-world corporate IT infrastructure. The primary objective was to deploy a Microsoft Active Directory (AD) Domain Controller, configure network services (DNS, DHCP, NAT) for connectivity, and automate user onboarding using PowerShell. The project concluded with a successful verification of a Windows 10 client joining the domain and authenticating with a centrally managed user account.

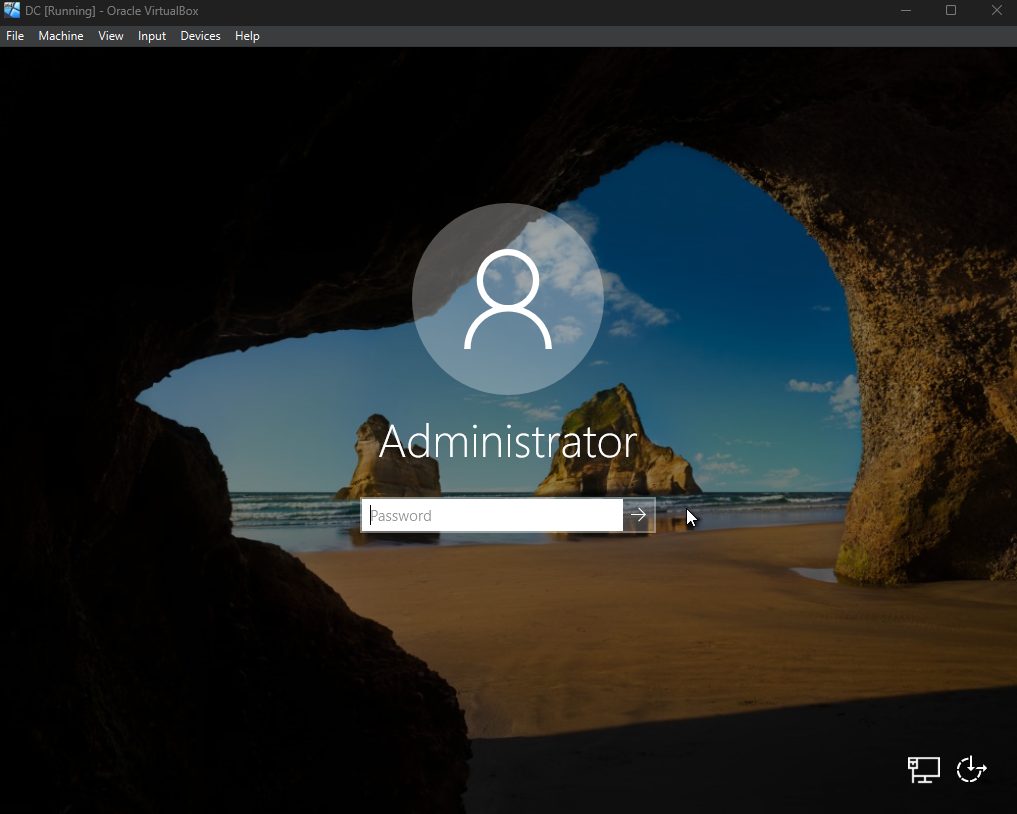
**2. Technical Environment**

* **Hypervisor:** Oracle VirtualBox
* **Server OS:** Windows Server 2019 (Domain Controller)
* **Client OS:** Windows 10 Enterprise
* **Tools:** PowerShell ISE, Remote Access Management Console

**2. Implementation & Screenshot Evidence**

**Phase 1: Server Configuration & Domain Controller Setup**

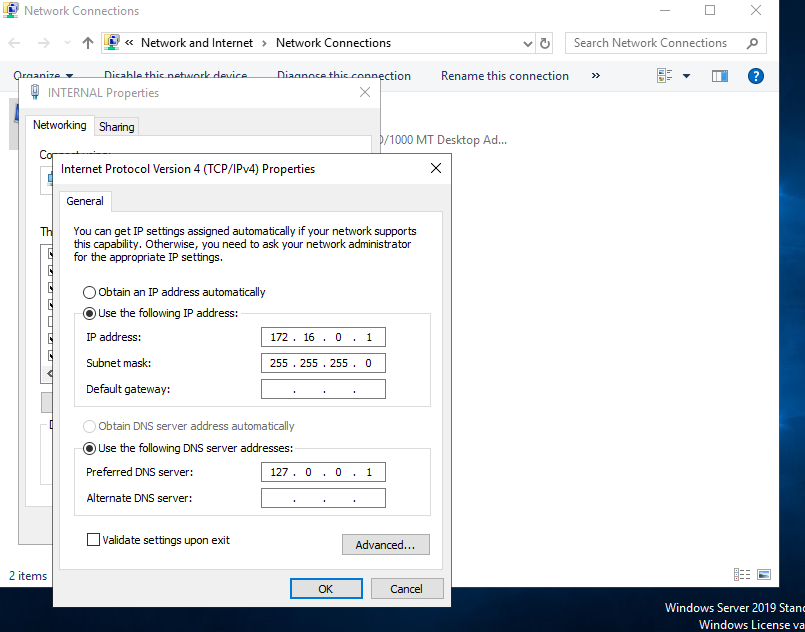
* The first phase focused on preparing the Windows Server 2019 environment. A static IP address was essential to ensure the server could reliably function as a DNS and Authentication server for the network.



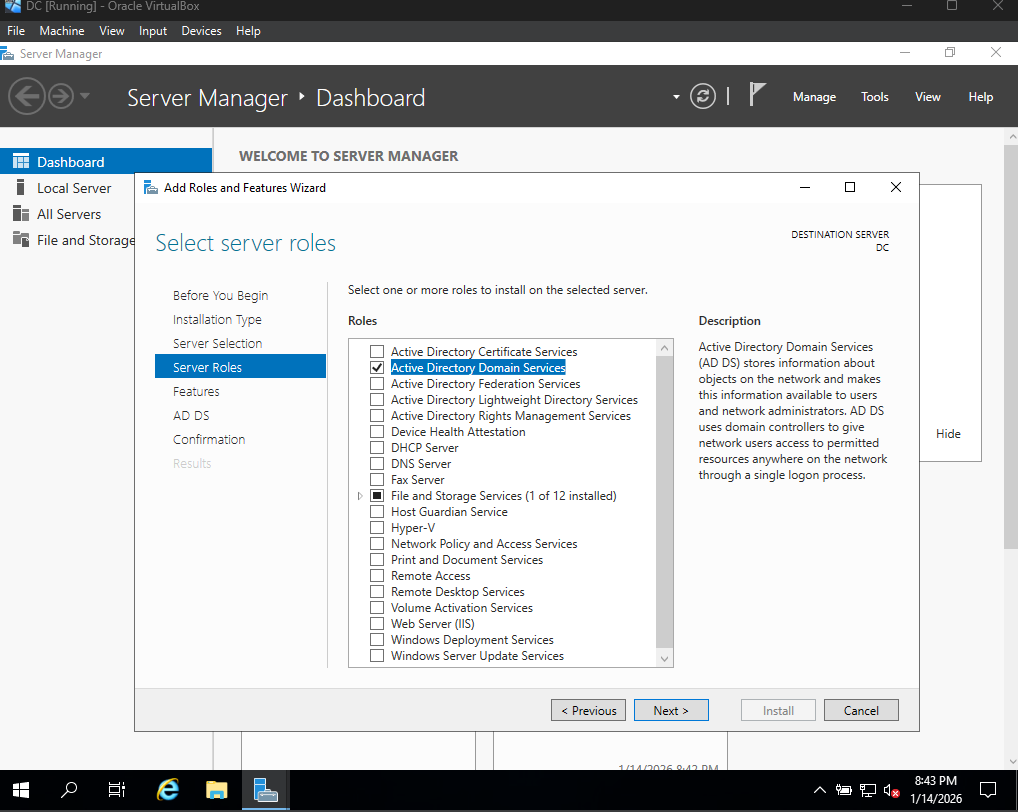
This shows the initial login to the Windows Server 2019 Desktop environment using the default **Administrator** account. This confirms the successful installation of the server operating system before any configuration began.



Here, I renamed the server from its default random string (e.g., WIN-P8...) to **"DC"** (Domain Controller). This is a critical step in enterprise management to ensure the server is easily identifiable on the network.



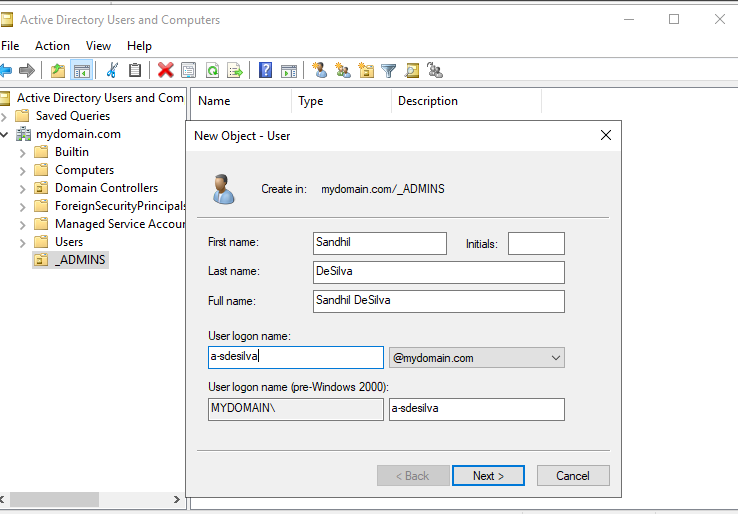
This screenshot displays the **IPv4 Configuration** for the server’s internal network adapter. I assigned the Static IP **172.16.0.1** and set the DNS server to 127.0.0.1 (localhost). This ensures the server looks to itself for DNS resolution, which is required for Active Directory to function correctly.



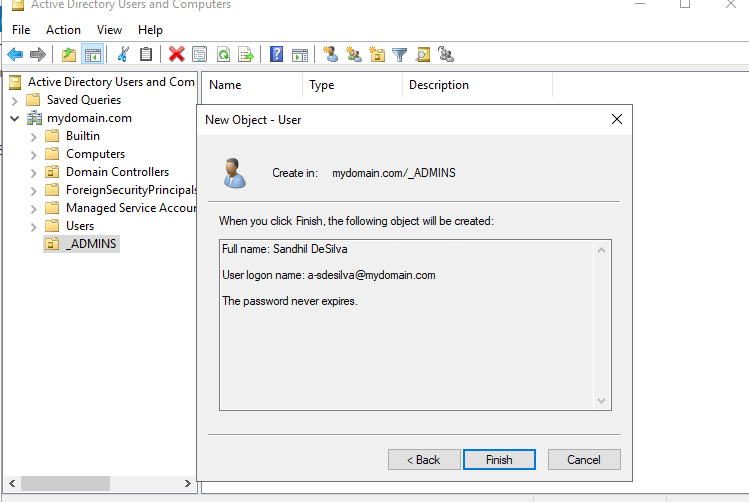
This shows the **Add Roles and Features Wizard**, where I selected **Active Directory Domain Services (AD DS)**. Installing this role is what promotes a standard server into a Domain Controller, allowing it to manage identities and computers.

**Phase 2: User Account Management**

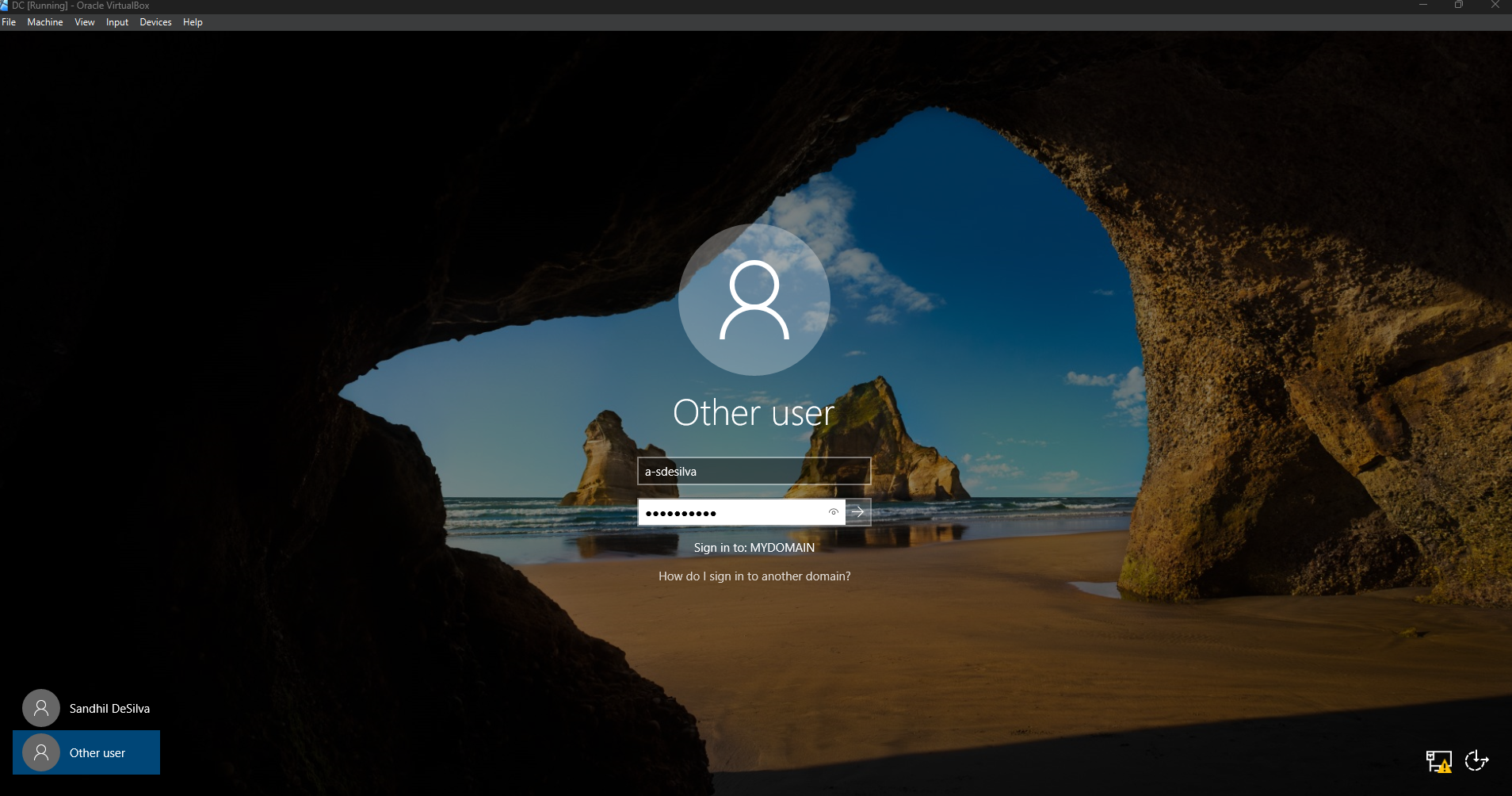
* Once the domain mydomain.com was established, I moved to managing user accounts. Best practice dictates not using the default Administrator account for daily tasks, so a new Admin account was created.



I used the **Active Directory Users and Computers (ADUC)** tool to create a new organizational user. This screenshot shows the creation of the admin user **"Sandhil DeSilva"** with the logon name a-sdesilva. The a- prefix is a common naming convention for "Admin" accounts.

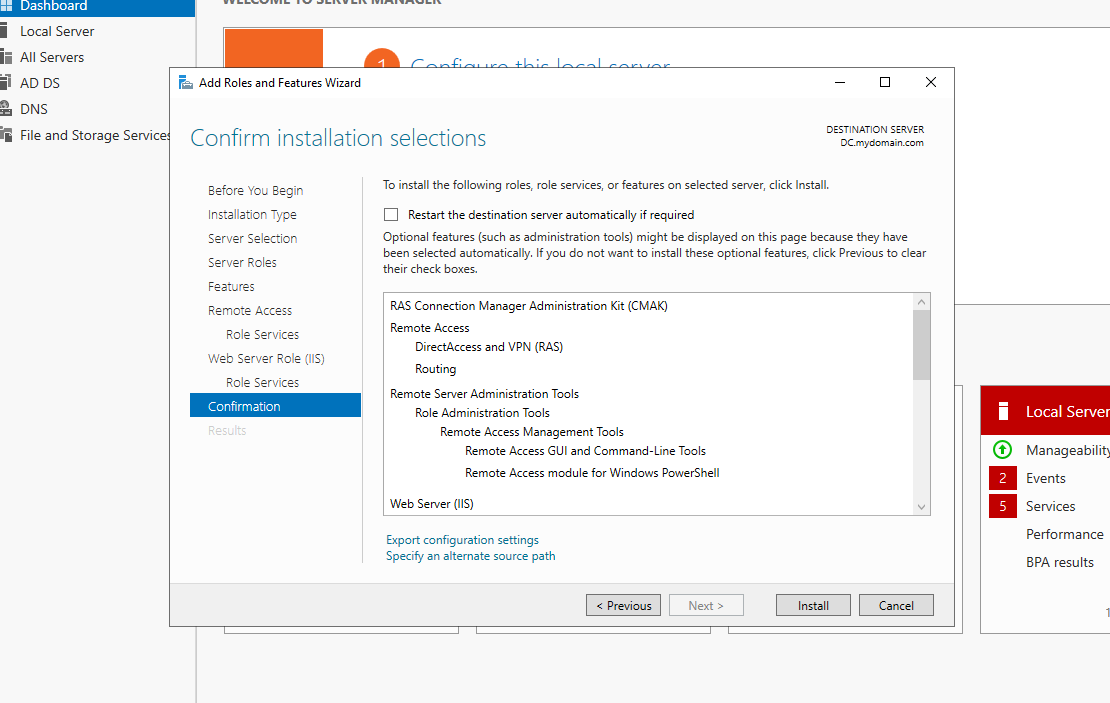


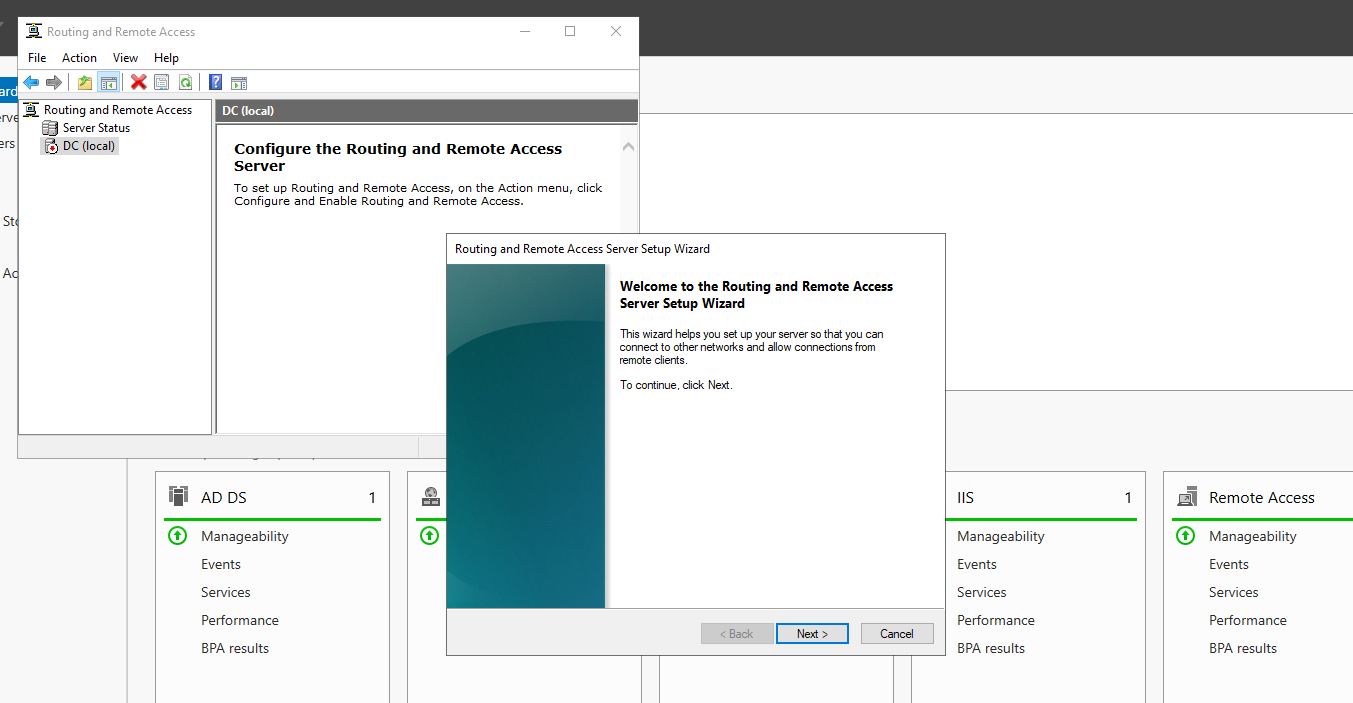
confirms the user object configuration before finalising. It shows the account is set to "Password never expires," which is typical for service or admin accounts in a lab environment to prevent lockout during testing.

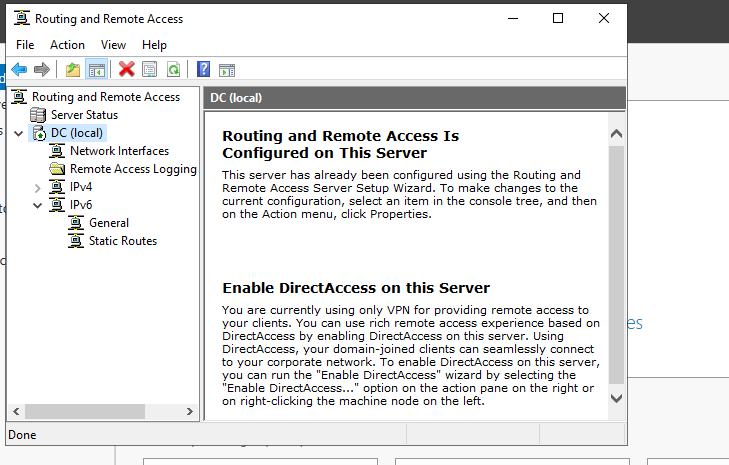
This verification step shows me successfully logging into the Domain Controller using the newly created **a-sdesilva** account. This proves that the Active Directory authentication database is active and working.

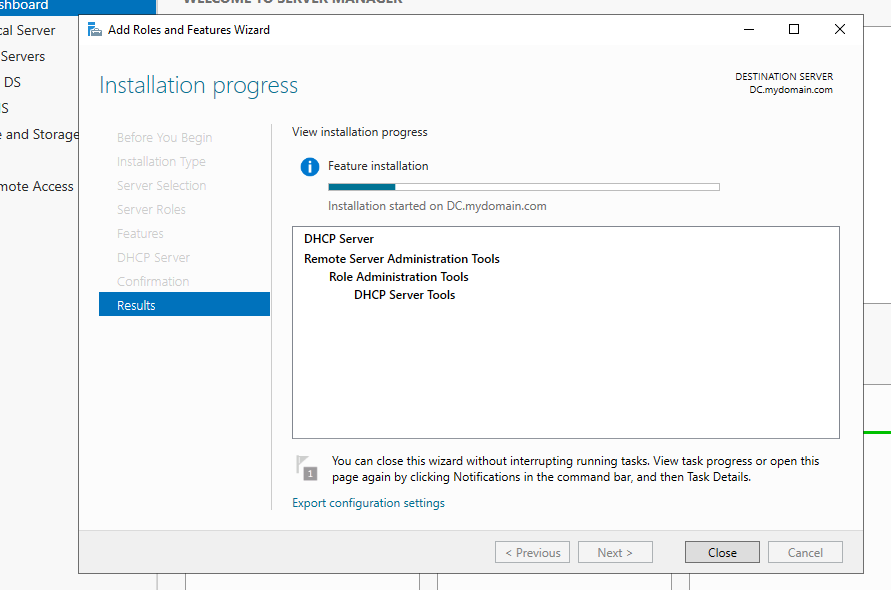
**Phase 3: Networking Services (NAT & DHCP)**

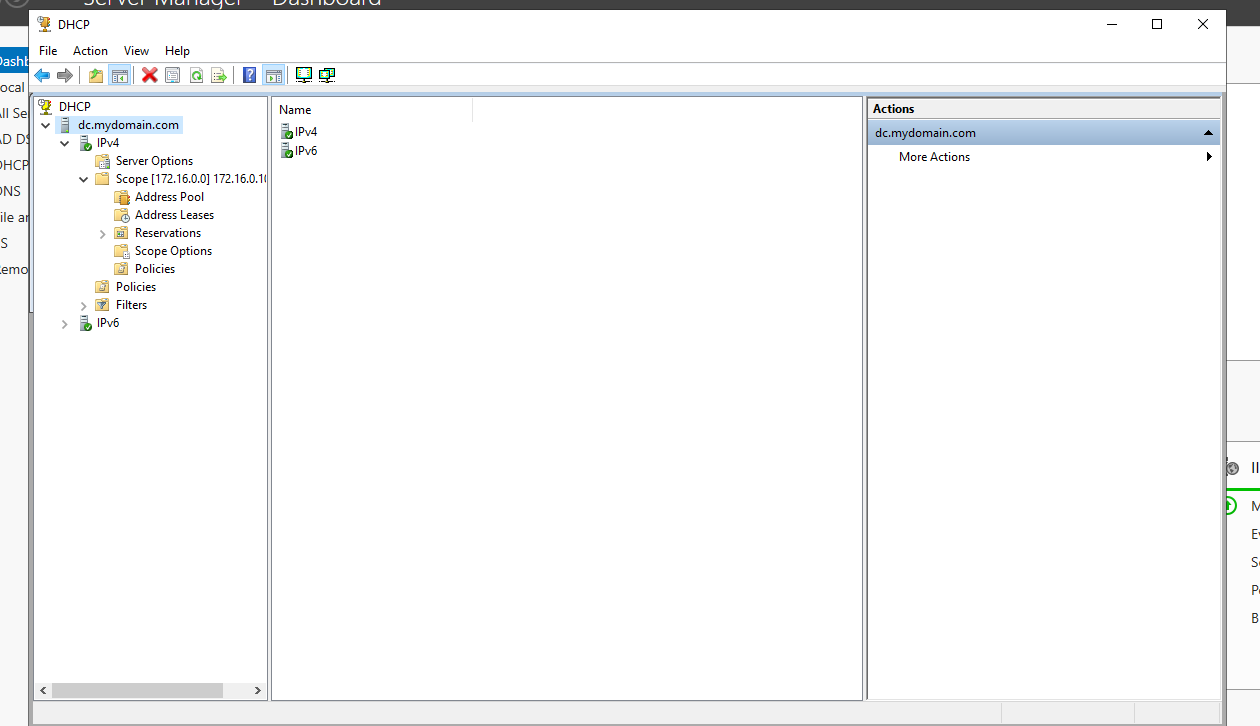
* To allow client computers on the private network to access the internet, I configured the server to act as a router using Network Address Translation (NAT).

This screenshot shows the installation of the **Remote Access** role. This role is required to configure routing and NAT, allowing the server to share its internet connection with other VMs on the private subnet.

I launched the **Routing and Remote Access Server Setup Wizard**. This wizard automates the configuration of the server's network interfaces to handle traffic routing between the internal private network and the external internet.

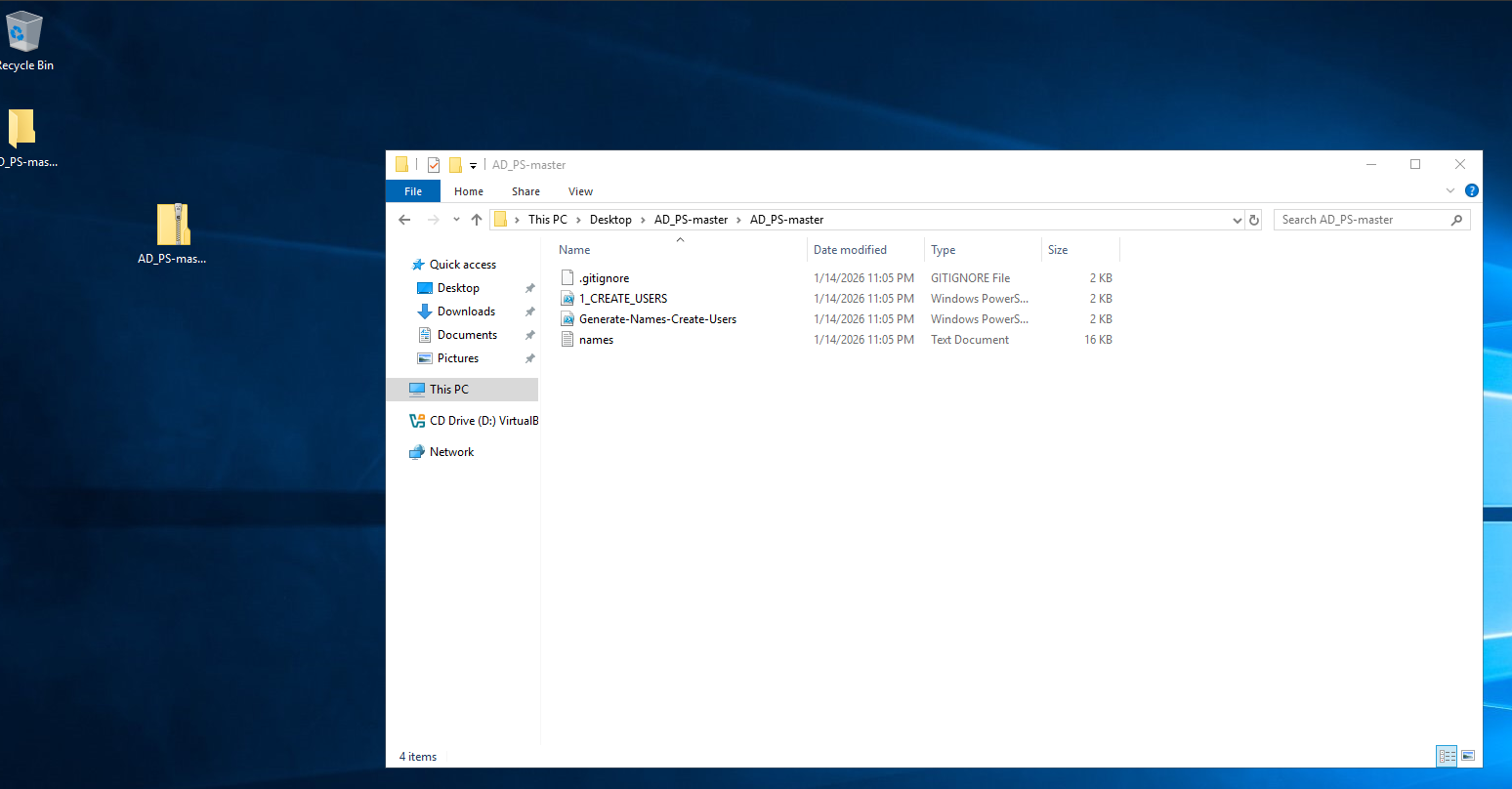
This shows the **Routing and Remote Access** console with a green status icon next to "DC (local)". This indicates that the NAT service is currently running and active, ready to route traffic for clients.

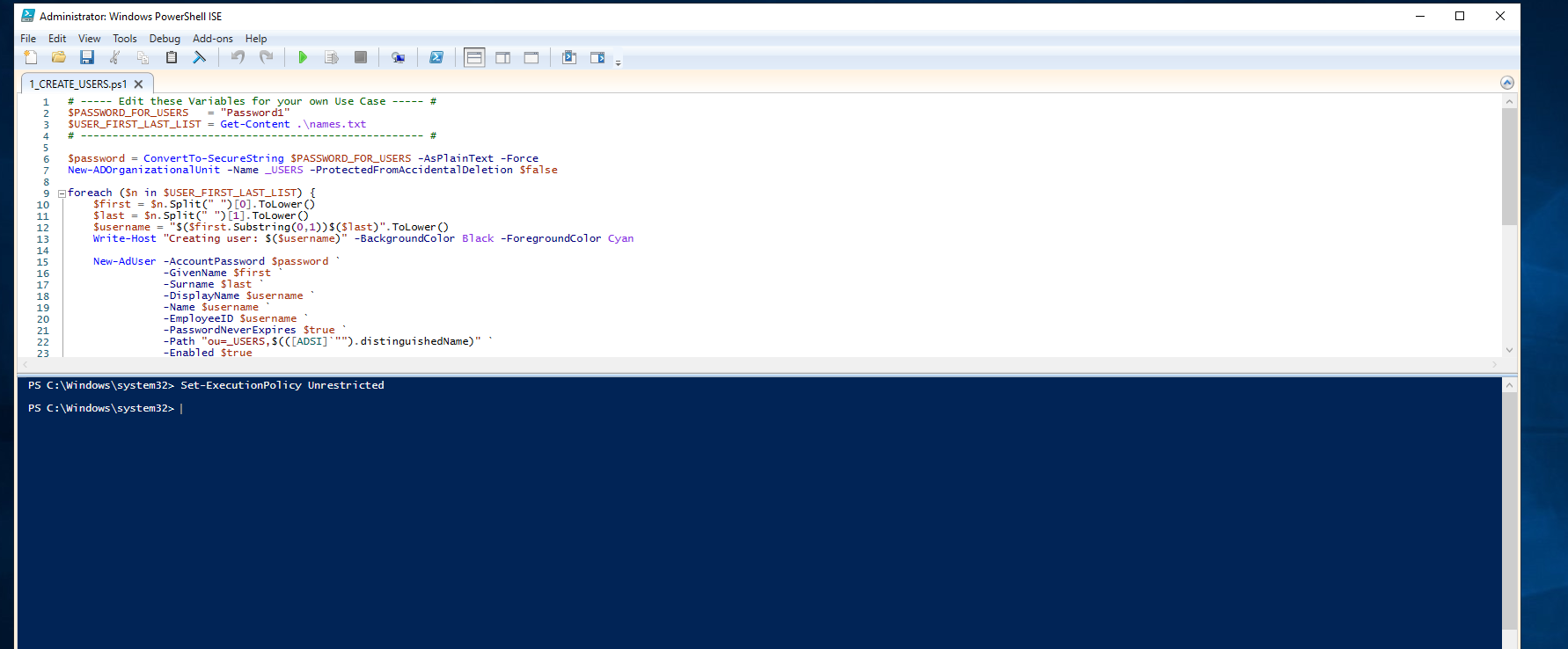
Here I am installing the **DHCP Server** role. DHCP is vital for automatically assigning IP addresses to client computers so that I do not have to manually configure every new machine that joins the network.

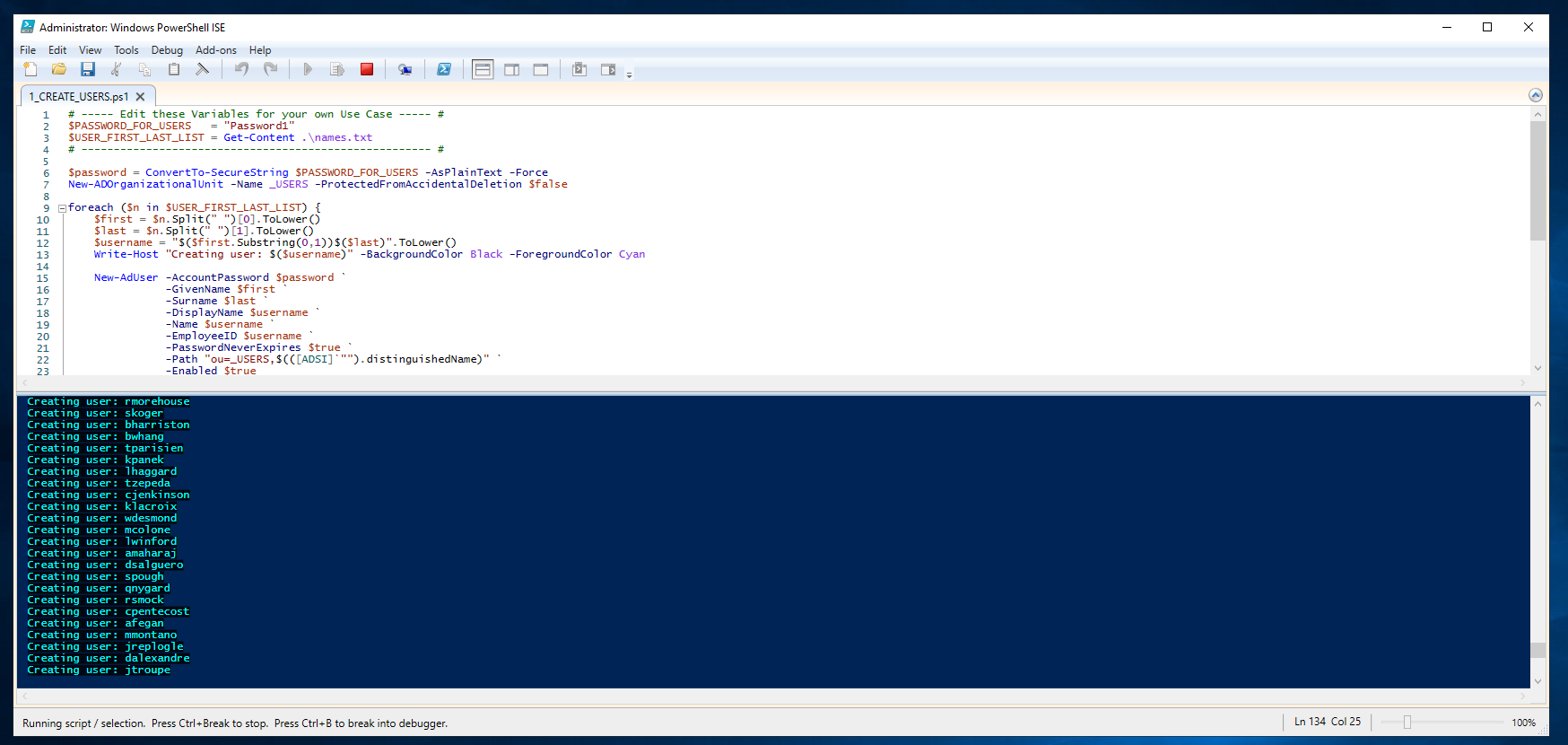
This shows the configured **DHCP Scope** within the DHCP Manager. The scope is set to distribute IP addresses in the 172.16.0.x range, ensuring that any client connecting to the network automatically receives a valid IP and network configuration.

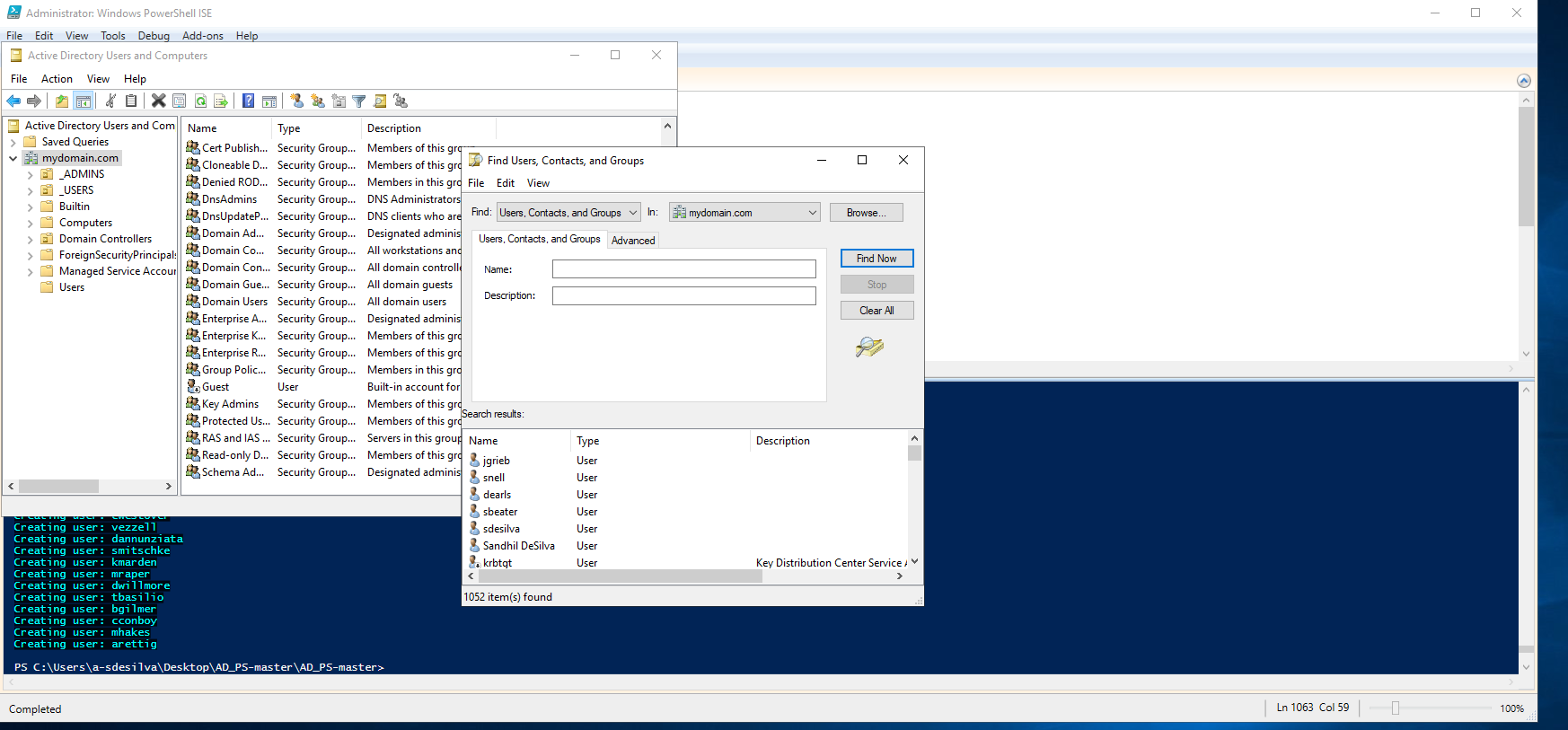
**Phase 4: Automation with PowerShell**

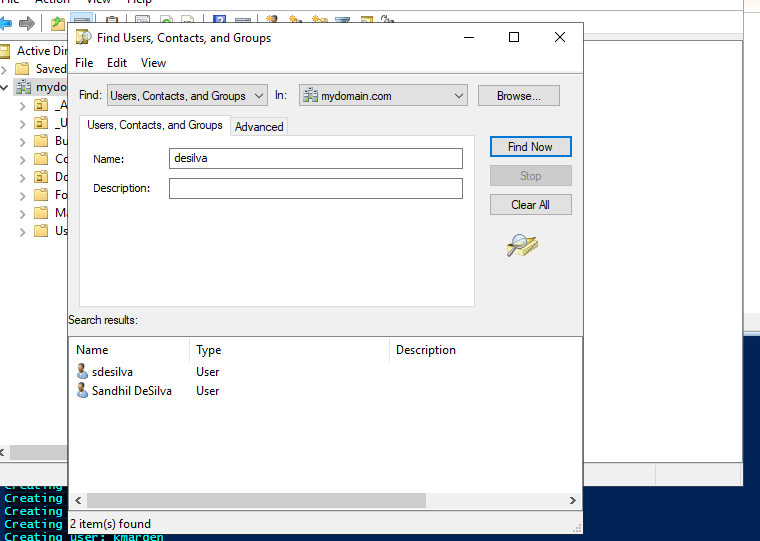
* To simulate a real-world bulk hiring scenario, I used a PowerShell script to automatically generate over 1,000 user accounts.

This shows the file structure for the automation project. The folder contains the **names.txt** file (the source data) and the **1\_CREATE\_USERS.ps1** script. This organization ensures the script can easily find the input file.

This is the **PowerShell ISE** displaying the script code. The script reads the text file and uses a foreach loop to run the New-ADUser command for every name in the list, setting a default password and adding them to the \_USERS OU.

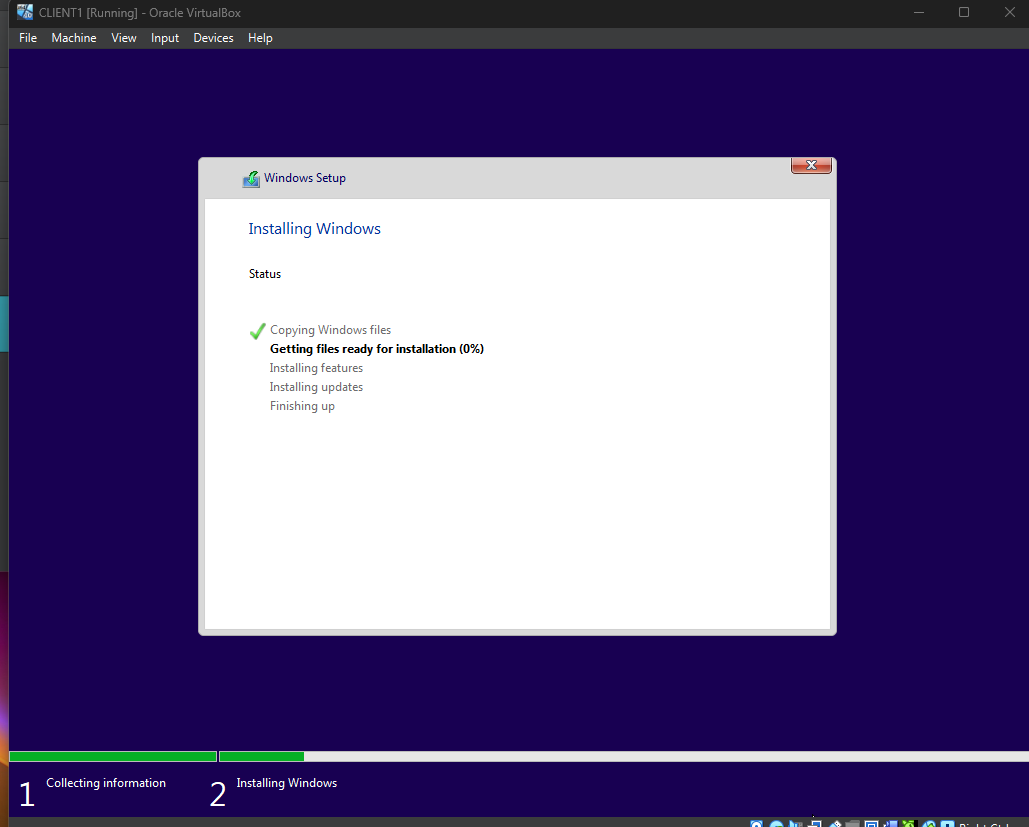
This screenshot captures the script **in execution**. The blue console output "Creating user: ..." confirms that the loop is running successfully and generating accounts in real-time without errors.

After the script finished, I used the search function in Active Directory to verify the results. The screenshot shows **1,052 items found**, proving that the bulk creation was successful and saved hours of manual data entry.

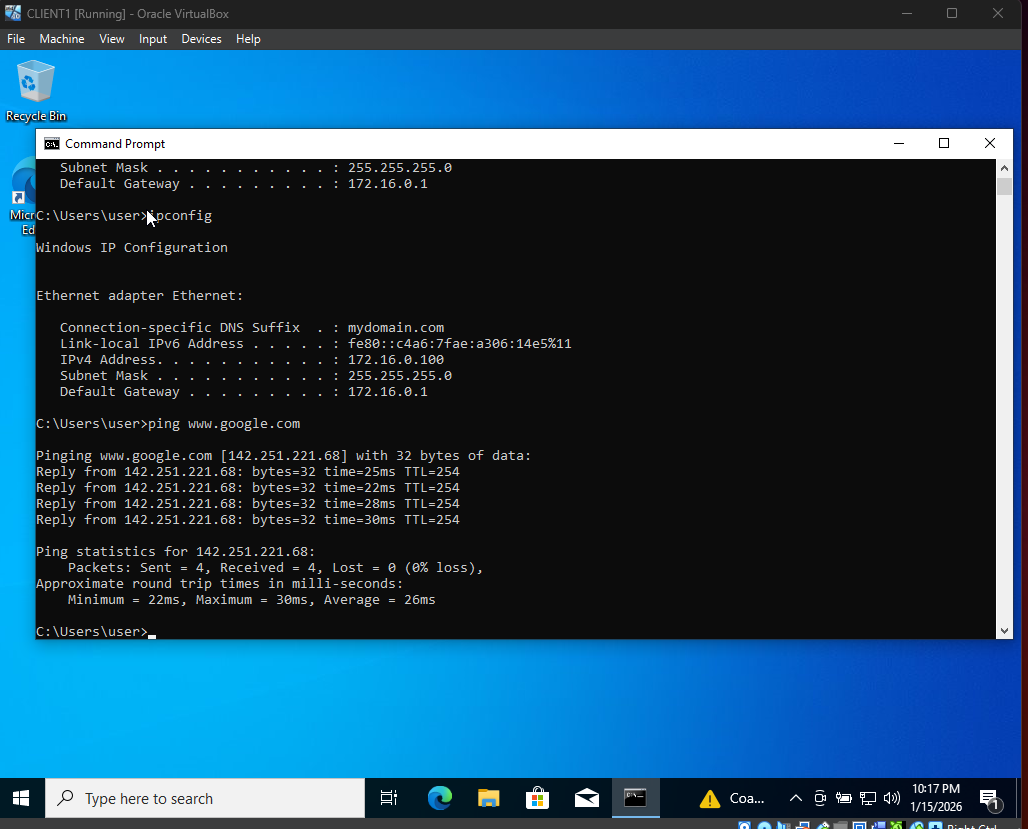
I performed a specific search for my own user account ("desilva") within the database. This verifies that individual accounts were created with the correct attributes and are searchable within the directory.

**Phase 5: Client Deployment & Testing**

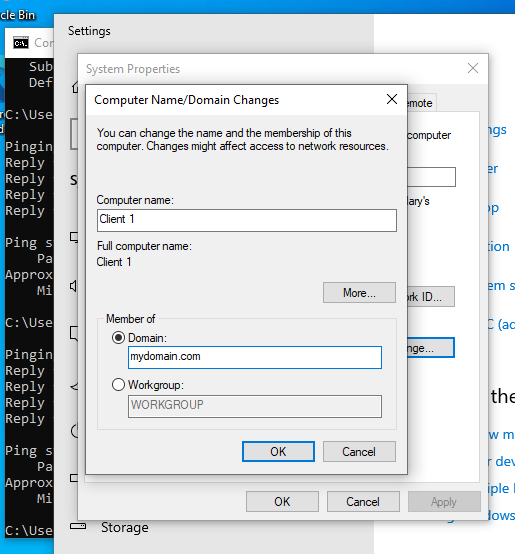
* The final phase involved setting up a Windows 10 "Client" machine to act as an employee workstation and joining it to the domain.

**[INSERT SCREENSHOT 18 HERE]** **Explanation:** 

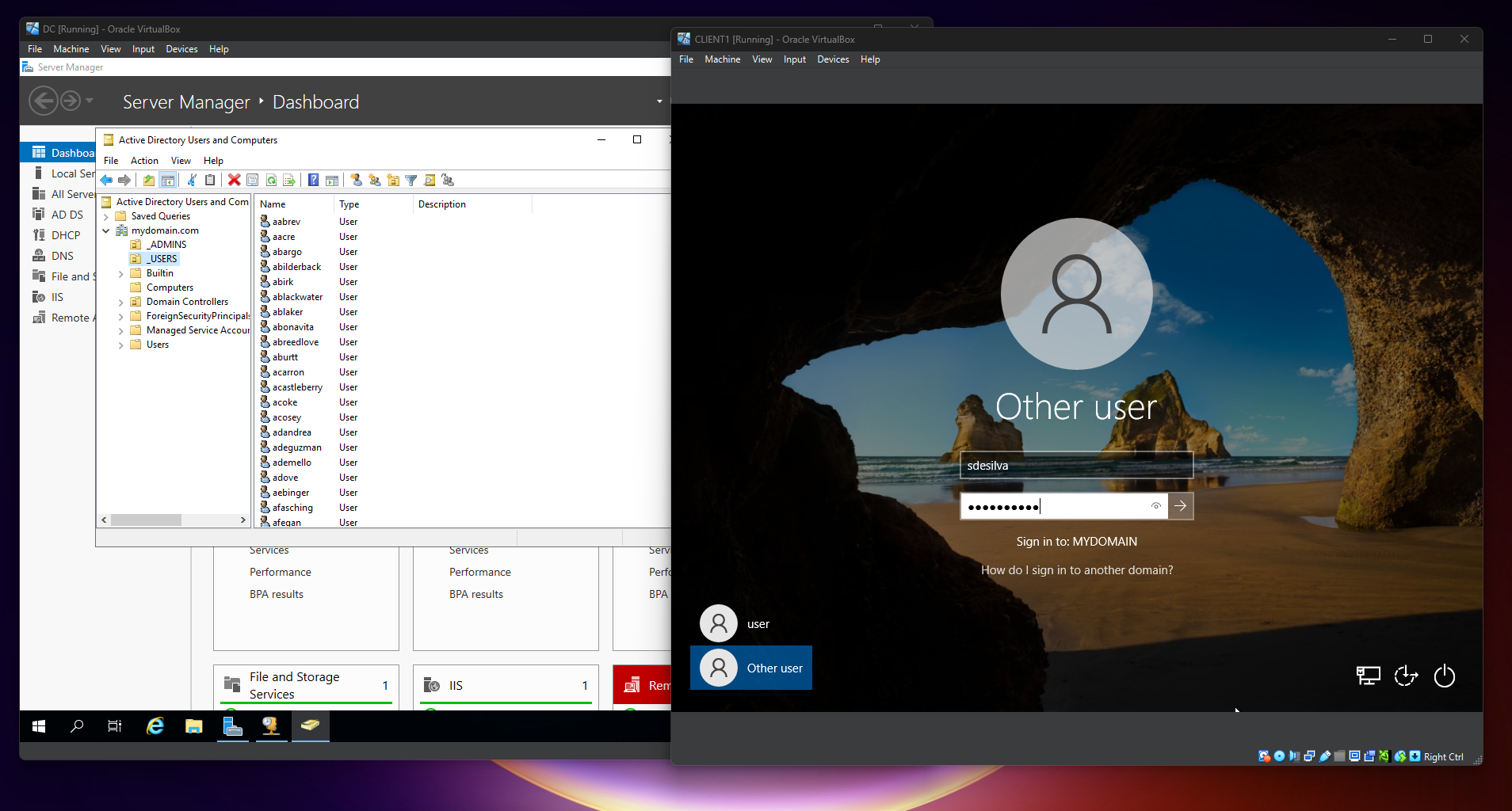
This shows the **Windows 10 Enterprise** installation process on the Client Virtual Machine. I used the Enterprise version because the "Home" version of Windows does not support joining a Domain.



This is a critical connectivity test. The command prompt shows ipconfig confirming the client received the IP **172.16.0.100** (from our DHCP server) and a successful **Ping to** [**www.google.com**](https://www.google.com). This proves the Server's NAT and DNS forwarding are working perfectly.



This screenshot shows the "System Properties" window where I successfully changed the computer's membership from "Workgroup" to **Domain: mydomain.com**. This is the moment the client computer officially became part of the managed corporate network.

After restarting the client, the login screen now allows me to sign in to **"MYDOMAIN"**. I am entering the credentials for the user sdesilva, which validates that the client is authenticating against the Domain Controller, not its local database.



The final screenshot shows the Windows 10 Desktop successfully logged in. The clean desktop environment confirms that a centrally managed domain user can log in to a workstation they have never used before, completing the project objectives.

**3. Conclusion**

This project successfully built a fully functional Enterprise Active Directory environment from scratch.

**Project Outcomes:**

* **Server Success:** A Windows Server 2019 Domain Controller is now running and actively managing the network.
* **Network Connectivity:** The internal network (DNS, DHCP, NAT) is working correctly, allowing isolated machines to access the internet securely.
* **Automation:** I successfully used PowerShell to automatically create over **1,000 user accounts** in seconds, replacing hours of manual work.
* **Client Verification:** A Windows 10 Client successfully joined the domain, proving that users can log in with their central credentials.

Overall, this lab proves that the infrastructure is fully operational and I can effectively set up and manage a professional IT environment.