

NOSS – Coursework 2

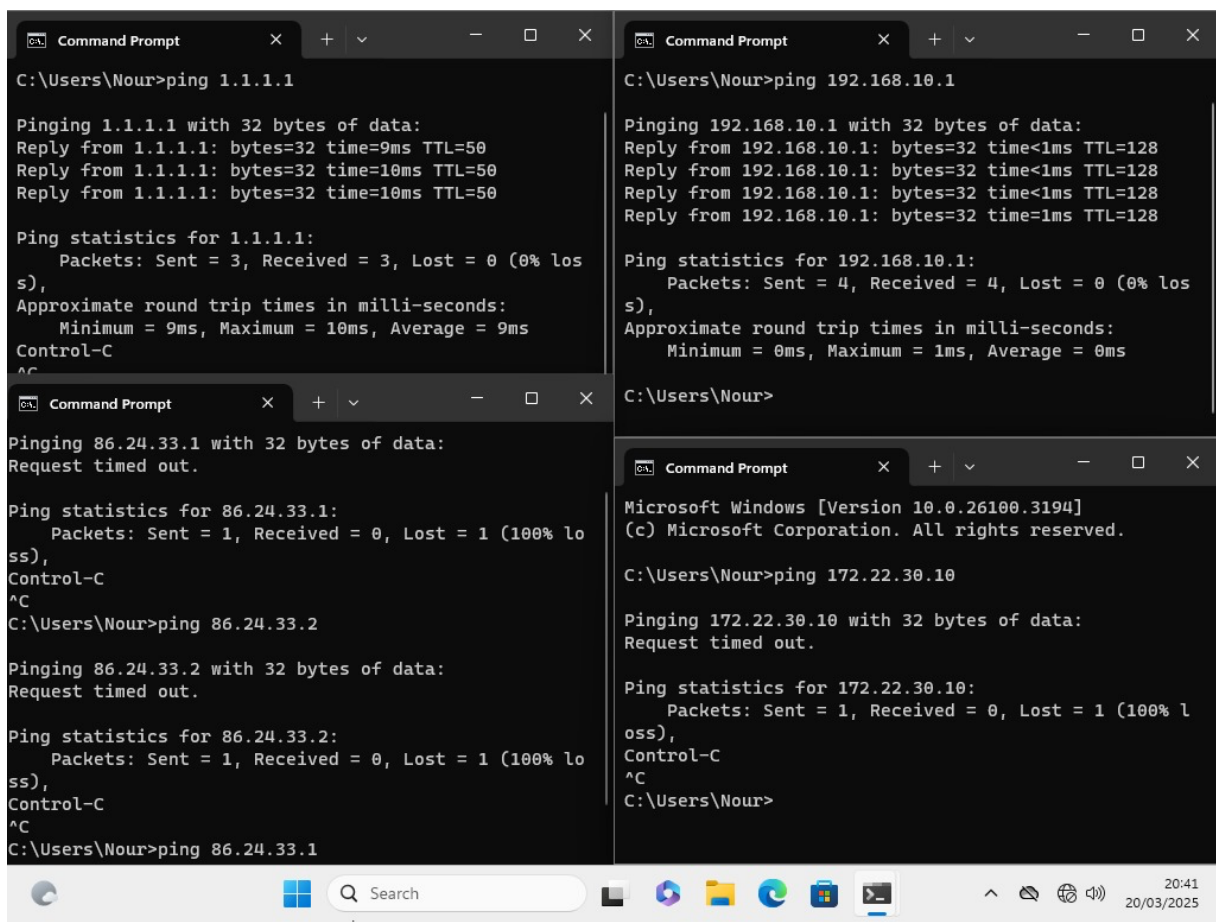
Introduction

This report is the continuation of Coursework 1. In this Coursework 2, we will explore different steps needed to configure, secure and manage a virtualized network environment. We will therefore understand key concepts such as VPN,DHCP, remote access and more. The aim of this coursework is to develop a robust and secure network infrastructure that is fully operational.

I- Swansea Network

A- Initial Connectivity Verification (Coursework 1)

Before implementing the new services, it may be useful to highlight that all systems are connected to the corresponding systems they are meant to communicate with. All ping tests were successfully completed thanks to the configurations made in Coursework 1.



The figure displays four screenshots of Windows Command Prompt windows, arranged in a 2x2 grid, showing the results of ping tests. The top-left window shows a successful ping to 1.1.1.1 with 32 bytes of data, receiving three replies with times of 9ms, 10ms, and 10ms, and a TTL of 50. The top-right window shows a successful ping to 192.168.10.1 with 32 bytes of data, receiving four replies with times less than 1ms and a TTL of 128. The bottom-left window shows a failed ping to 86.24.33.1 with 32 bytes of data, resulting in a request timed out. The bottom-right window shows a failed ping to 172.22.30.10 with 32 bytes of data, resulting in a request timed out. The Windows taskbar at the bottom shows the system clock as 20:41 on 20/03/2025.

```
C:\Users\Nour>ping 1.1.1.1

Pinging 1.1.1.1 with 32 bytes of data:
Reply from 1.1.1.1: bytes=32 time=9ms TTL=50
Reply from 1.1.1.1: bytes=32 time=10ms TTL=50
Reply from 1.1.1.1: bytes=32 time=10ms TTL=50

Ping statistics for 1.1.1.1:
    Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 9ms, Maximum = 10ms, Average = 9ms
Control-C
^C

C:\Users\Nour>ping 192.168.10.1

Pinging 192.168.10.1 with 32 bytes of data:
Reply from 192.168.10.1: bytes=32 time<1ms TTL=128
Reply from 192.168.10.1: bytes=32 time<1ms TTL=128
Reply from 192.168.10.1: bytes=32 time<1ms TTL=128
Reply from 192.168.10.1: bytes=32 time=1ms TTL=128

Ping statistics for 192.168.10.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
Control-C
^C

C:\Users\Nour>ping 86.24.33.1

Pinging 86.24.33.1 with 32 bytes of data:
Request timed out.

Ping statistics for 86.24.33.1:
    Packets: Sent = 1, Received = 0, Lost = 1 (100% loss),
Control-C
^C

C:\Users\Nour>ping 86.24.33.2

Pinging 86.24.33.2 with 32 bytes of data:
Request timed out.

Ping statistics for 86.24.33.2:
    Packets: Sent = 1, Received = 0, Lost = 1 (100% loss),
Control-C
^C

C:\Users\Nour>ping 86.24.33.1

Pinging 86.24.33.1 with 32 bytes of data:
Request timed out.

Ping statistics for 86.24.33.1:
    Packets: Sent = 1, Received = 0, Lost = 1 (100% loss),
Control-C
^C

C:\Users\Nour>ping 172.22.30.10

Pinging 172.22.30.10 with 32 bytes of data:
Request timed out.

Ping statistics for 172.22.30.10:
    Packets: Sent = 1, Received = 0, Lost = 1 (100% loss),
Control-C
^C

C:\Users\Nour>
```

Fig1 : Pinging from Windows 11

Fig2 : Pinging from Windows 2022

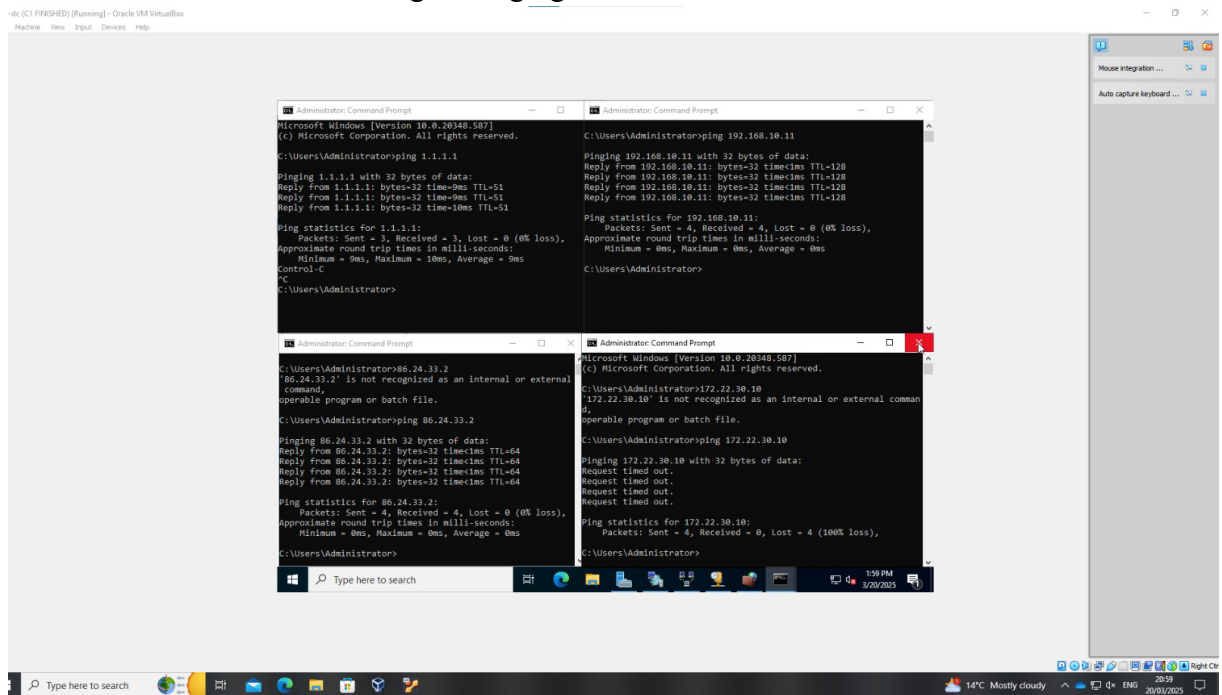


Fig3 : Pinging from Debian

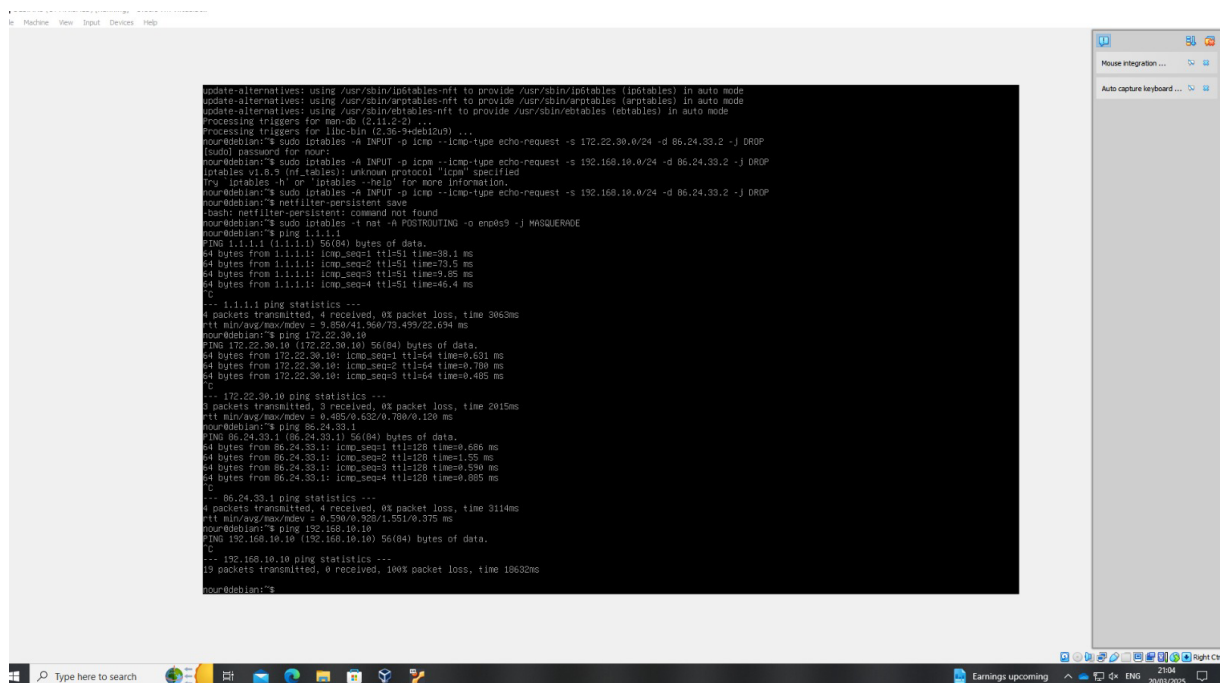
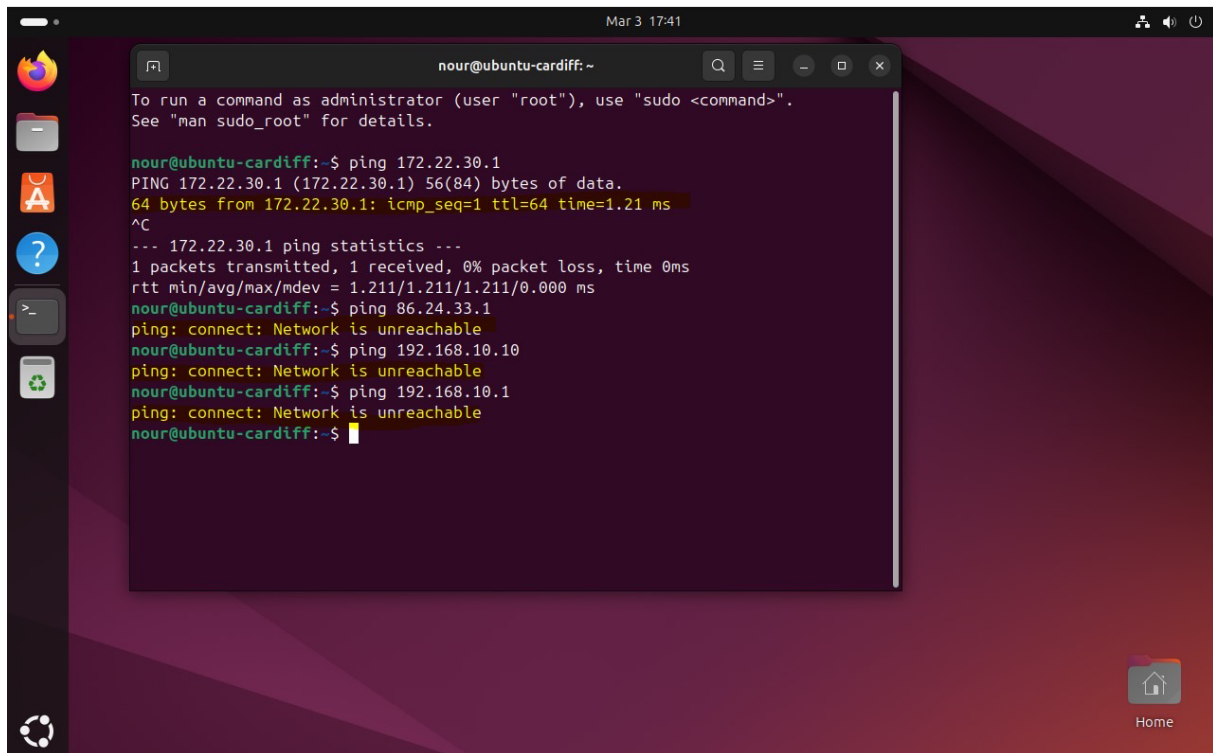


Fig4 : Pinging from Ubuntu

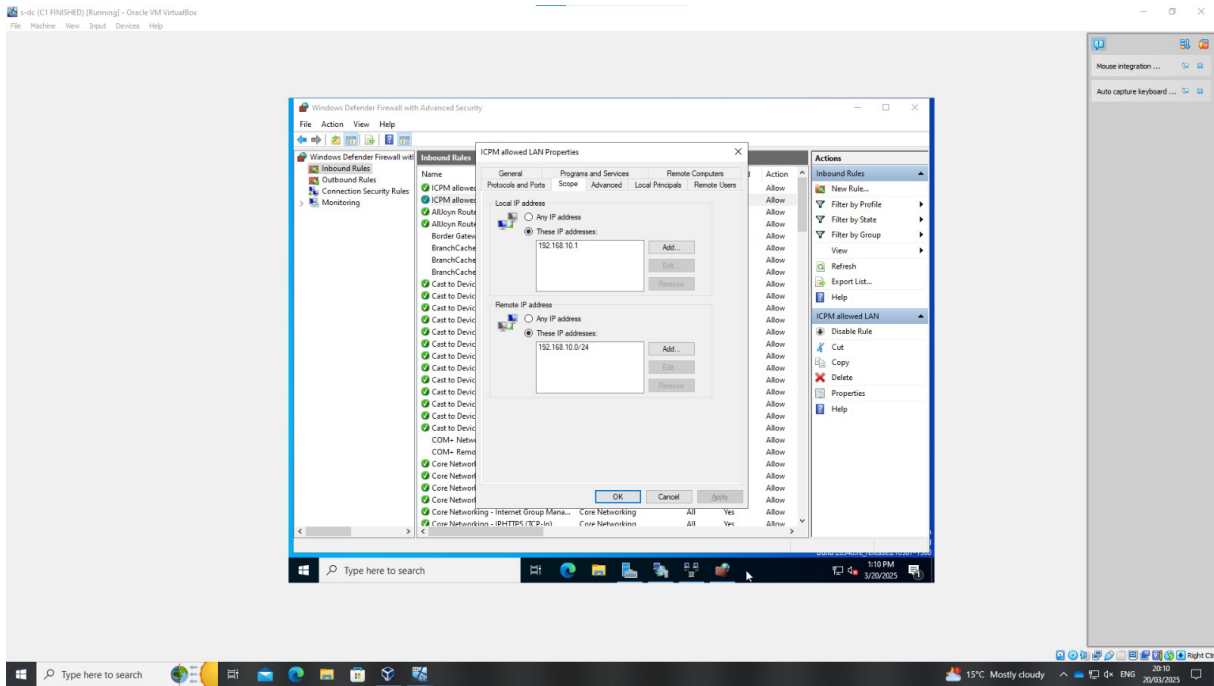


The image shows a terminal window on an Ubuntu desktop. The window title is 'nour@ubuntu-cardiff: ~'. The terminal output shows a successful ping to 172.22.30.1 and three failed pings to 86.24.33.1 and 192.168.10.10. The desktop background is a dark purple gradient. The left sidebar contains icons for Firefox, Files, Applications, Help, Terminal, and Recycle Bin. The bottom right corner has a 'Home' button.

```
nour@ubuntu-cardiff: ~  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
nour@ubuntu-cardiff:~$ ping 172.22.30.1  
PING 172.22.30.1 (172.22.30.1) 56(84) bytes of data:  
64 bytes from 172.22.30.1: icmp_seq=1 ttl=64 time=1.21 ms  
^C  
--- 172.22.30.1 ping statistics ---  
1 packets transmitted, 1 received, 0% packet loss, time 0ms  
rtt min/avg/max/mdev = 1.211/1.211/1.211/0.000 ms  
nour@ubuntu-cardiff:~$ ping 86.24.33.1  
ping: connect: Network is unreachable  
nour@ubuntu-cardiff:~$ ping 192.168.10.10  
ping: connect: Network is unreachable  
nour@ubuntu-cardiff:~$ ping 192.168.10.1  
ping: connect: Network is unreachable  
nour@ubuntu-cardiff:~$
```

B- Configuring Firewall Rules to Enable ICMP Traffic

Fig5 : Firewall Rules Reactivated LAN



We configured a rule that allows ICMP traffic between specific IP addresses only. This rule applies to the local host with IP address 192.168.10.1 and permits ICMP communication exclusively with the remote IP address 192.168.10.254. This configuration ensures that only trusted devices within the LAN can exchange ping request, which increase security by restricting ICMP access. We configured the same thing within the WAN.

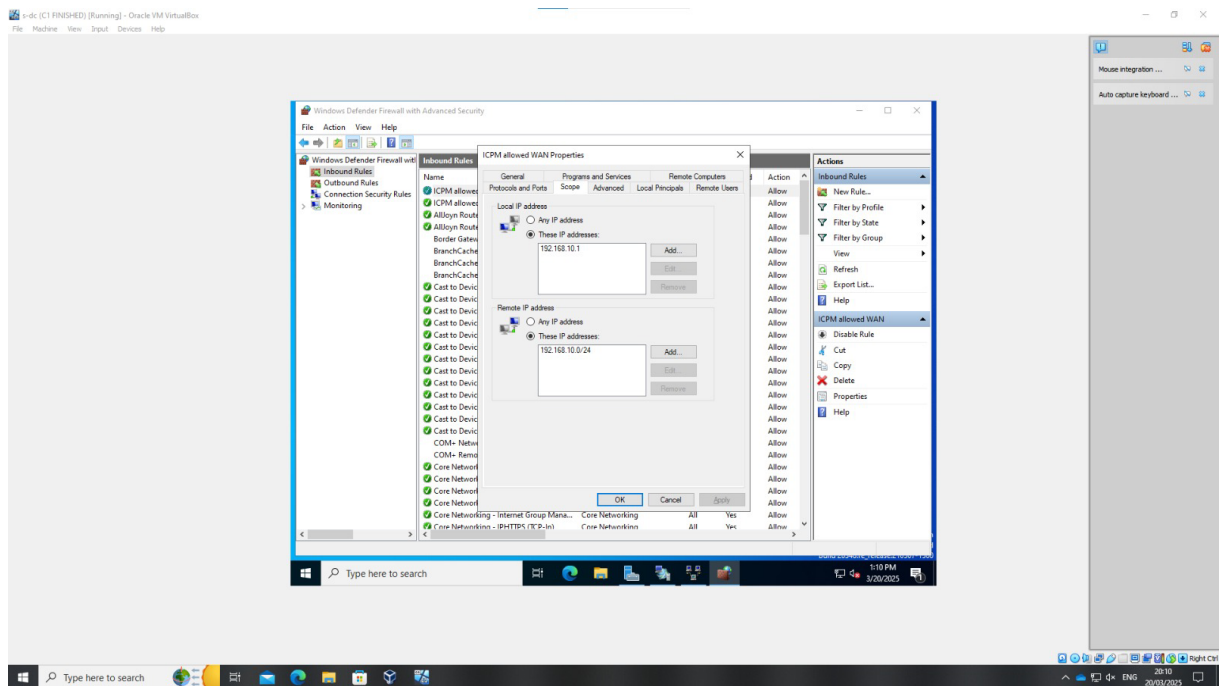


Fig6 : Firewall Rules Reactivated WAN

After that, we configured the Internet because our servers didn't have access to Internet, due to the failure of ping responses from 1.1.1.1

C- DNS

Then, I configured DNS as it allows computers to resolve domain names into IP addresses. That enables communication between device and domain services.

The configuration of this screenshot bellow restricts the DNS service to listen only on the IP address configured, rather than all available IP addresses.

This ensures that only authorized machines can query the DNS server, which is useful as it limits the DNS service to specific interfaces for security.

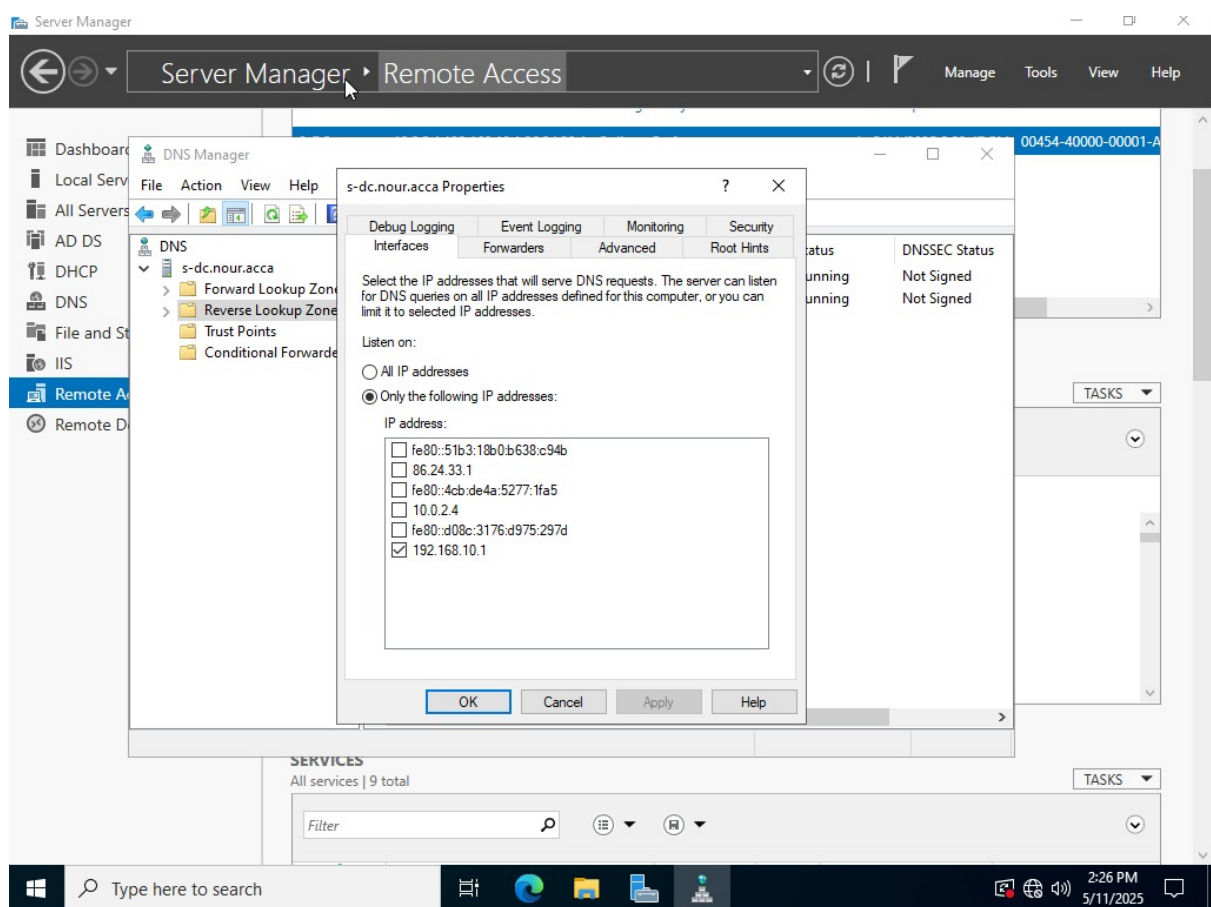


Fig7 : DNS Server IP Address Configuration

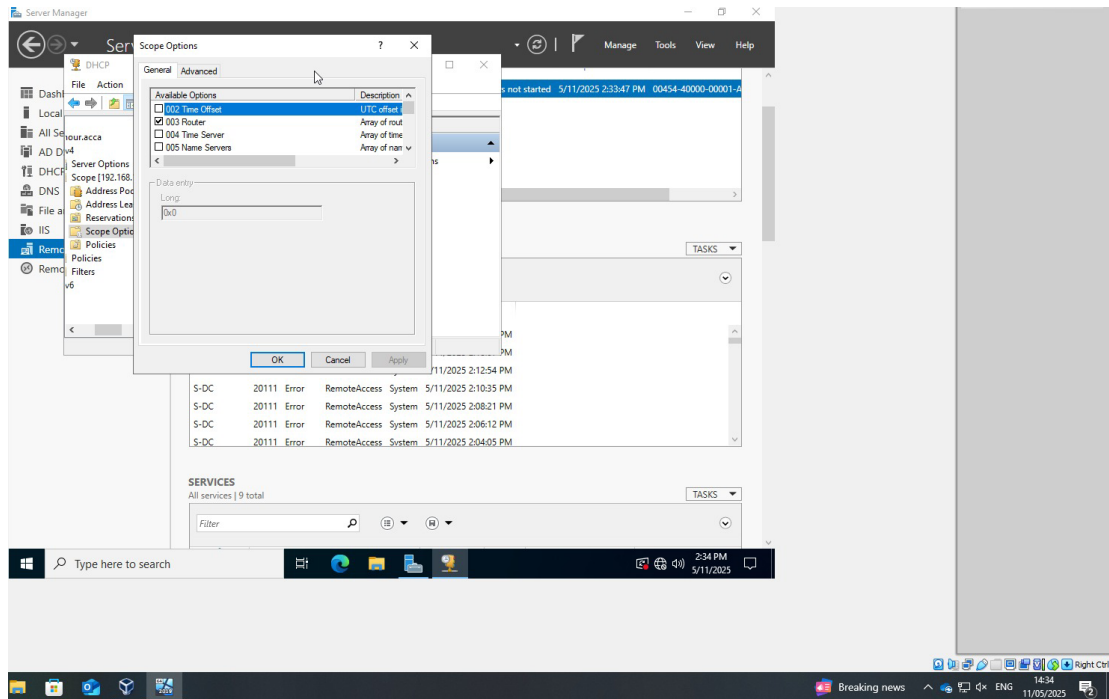


Fig8 : DHCP Scope Option Configuration – Setting DNS

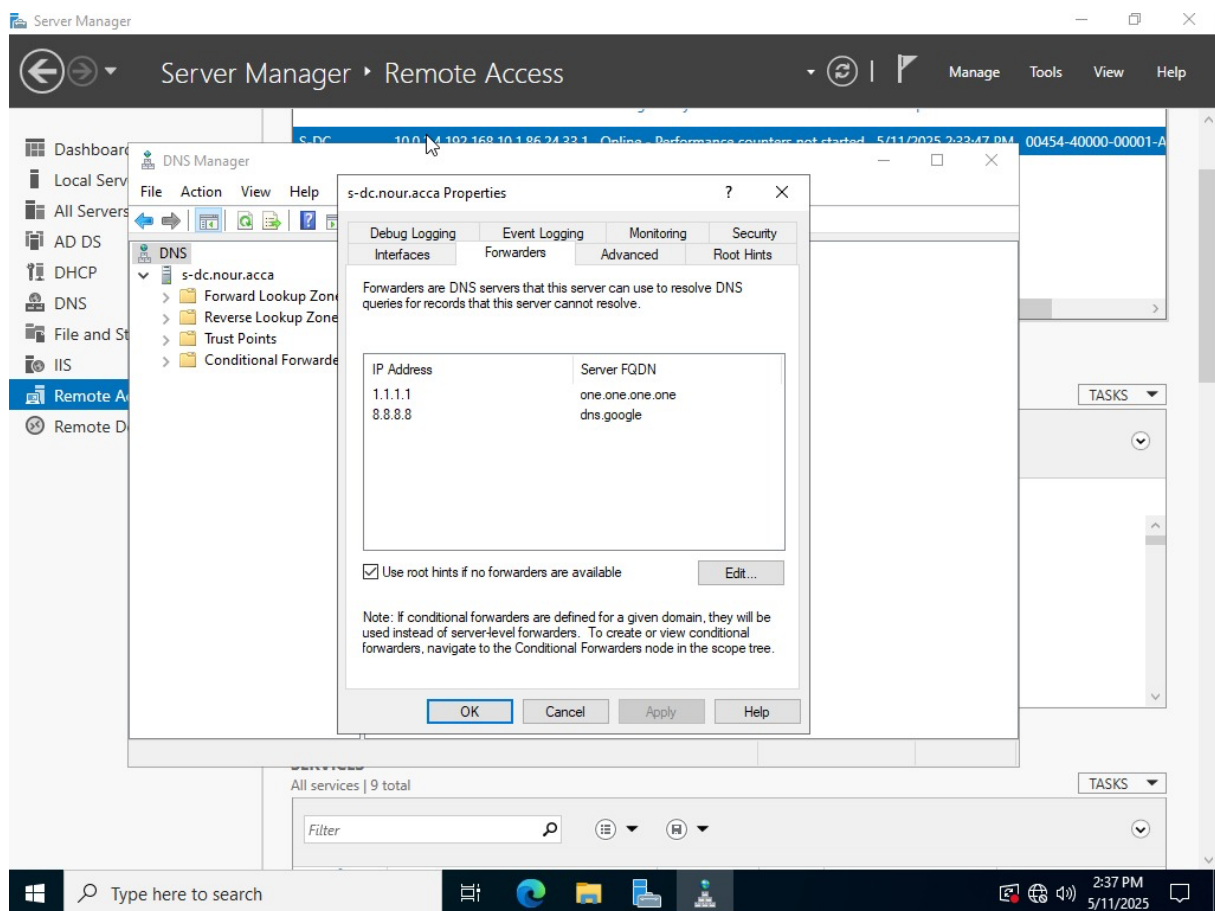


Fig9 : Configuring DNS Forwarders for External in DNS Manger

D- Active Directory Domain Services Setup

Then, I installed the Active Directory Domain Services (AD DS) role to the server using Server Manager. The required features for AD DS were pre-selected automatically during the setup process.

Then, in the same way we installed the AD DS, we install the DNS role.

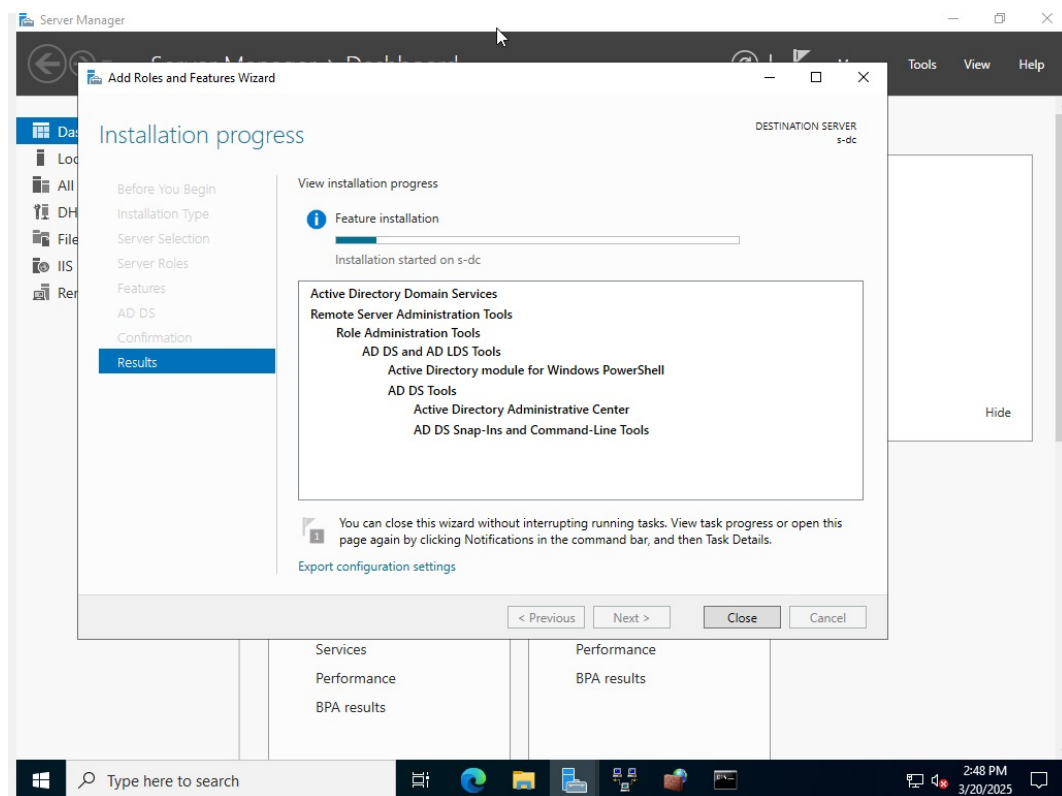


Fig10 : AD DS Installation Progress

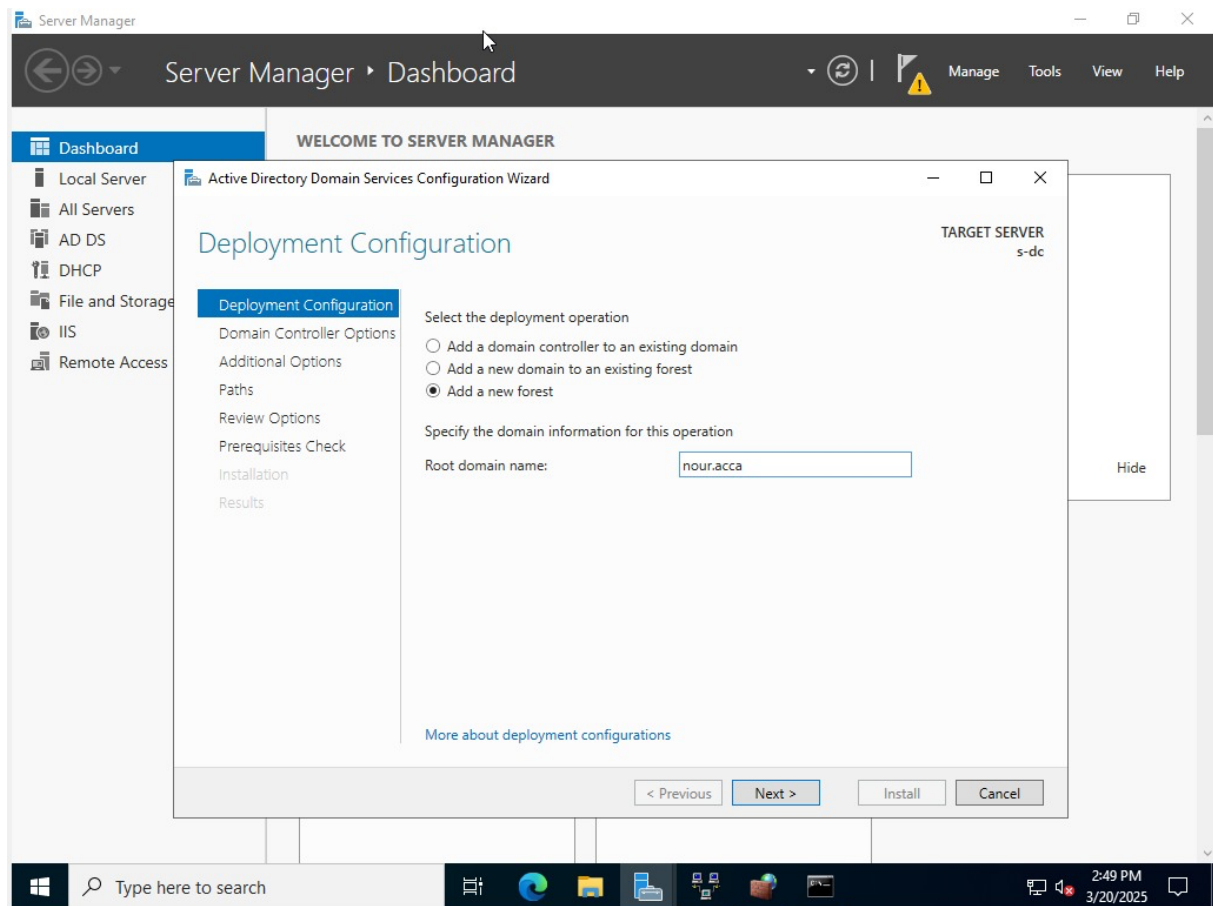


Fig11 : Forest Configuration

This step is crucial for our project because it creates a forest that is the foundation of our Active Directory environment. Without the creation of the forest, we would not be able to implement domain services, group policies, and establish trust between the different part of our network.

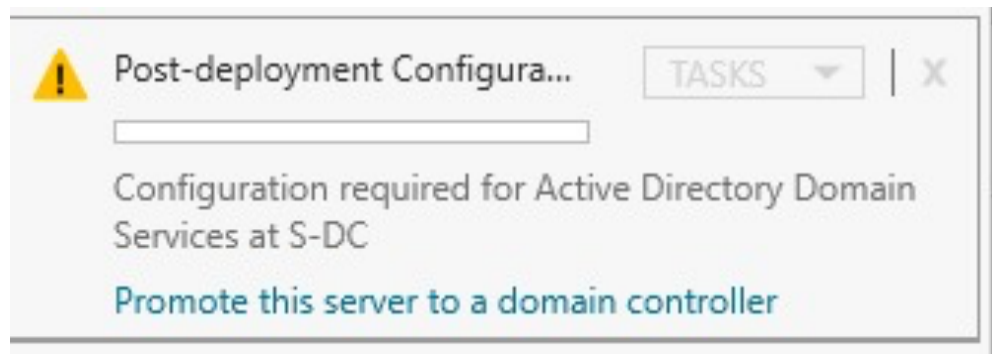


Fig12 :

Promote Server to Domain Controller

E- DHCP

The DHCP (Dynamic Host Configuration Protocol) allows the server to automatically assign IP addresses and other network configuration parameters to client devices.

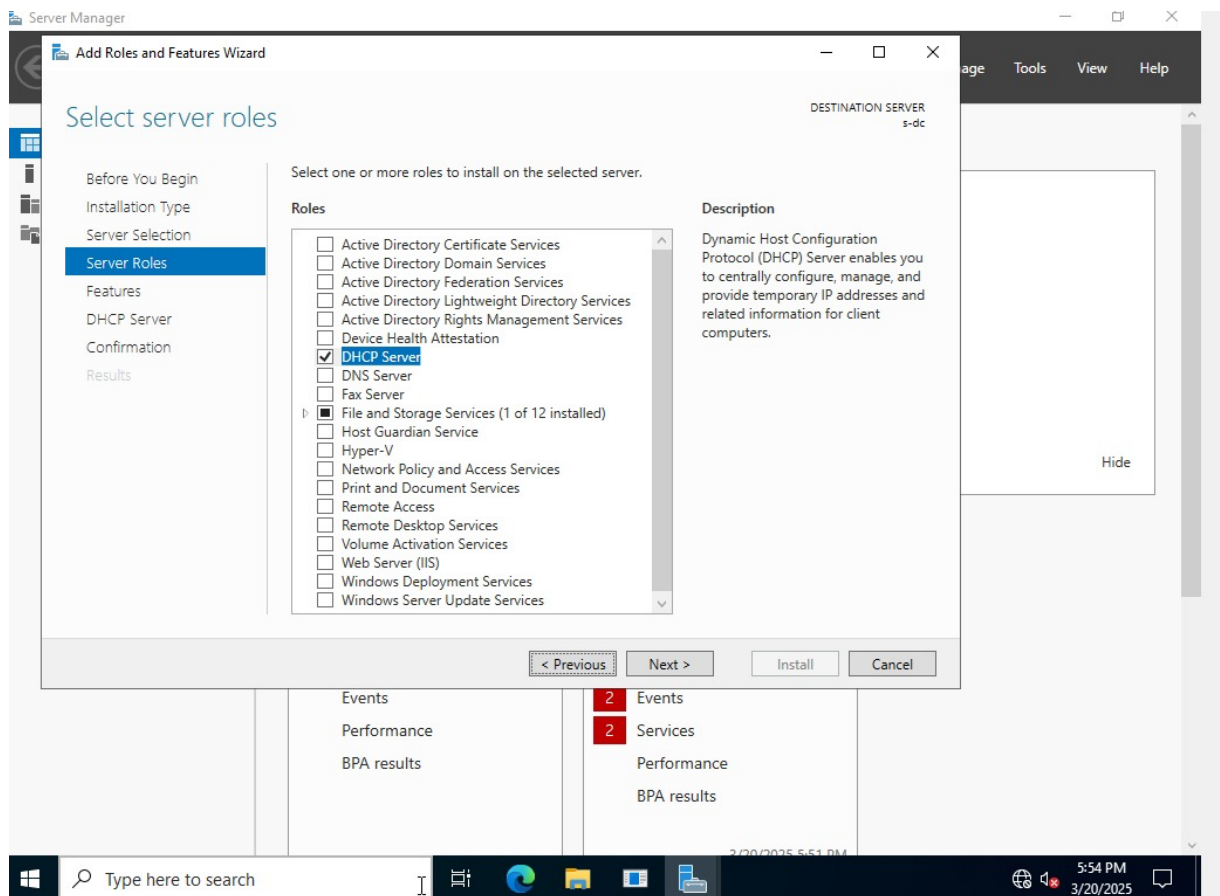


Fig13 : DHCP Installation

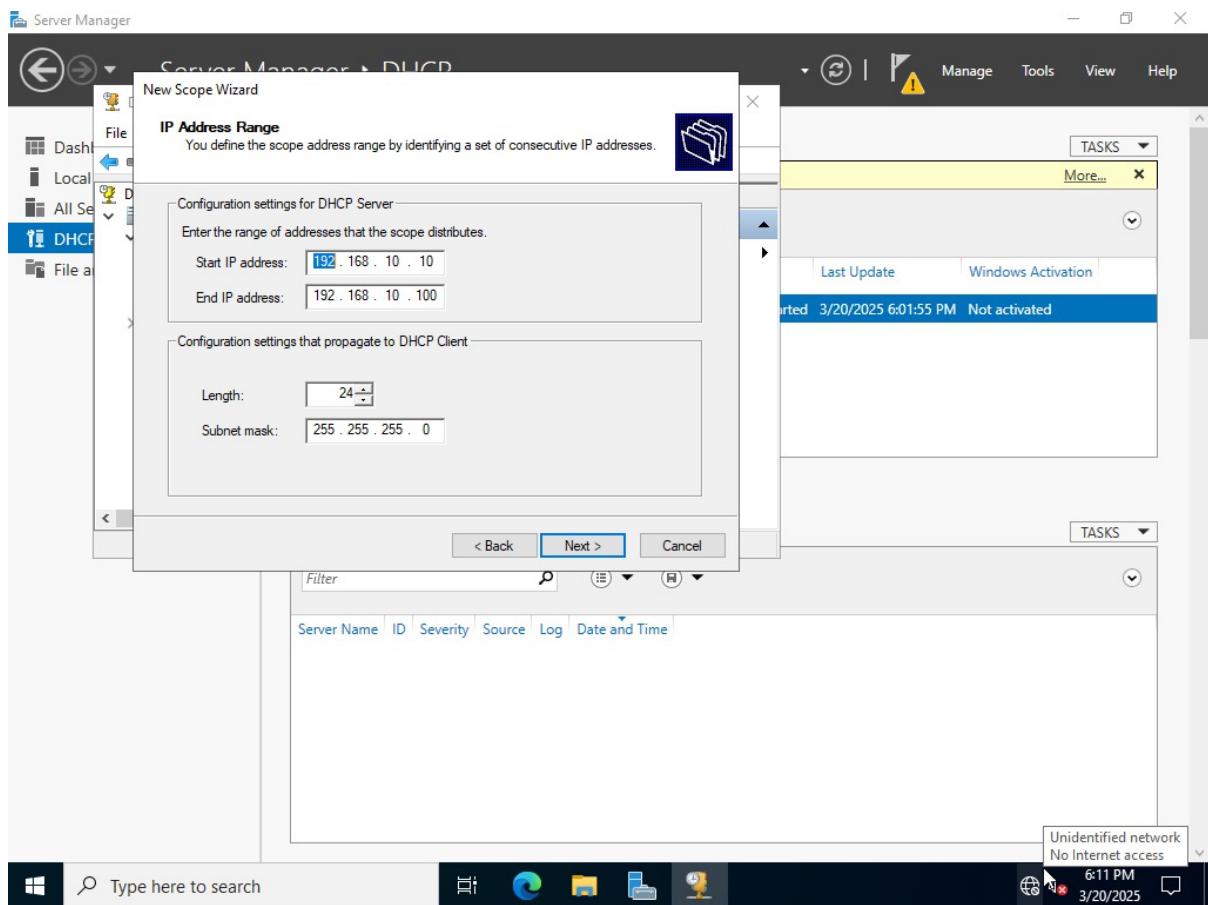


Fig14: IP addresses Configuration

This screenshot bellows illustrates the IP Address Range configuration step. The administrator is specifying the range of IP addresses that the DHCP server will allocate to client devices.

The DHCP server will automatically allocate IP addresses to client machines within the range of 192.168.10.10 to 192.168.10.100 on the network.

After completing the configuration, the client was tested to verify that it successfully obtained an IP address from the DHCP server.

By using the ipconfig command, we verified that the client received an IP address within the specified range, along with the appropriate subnet mask and default gateway.

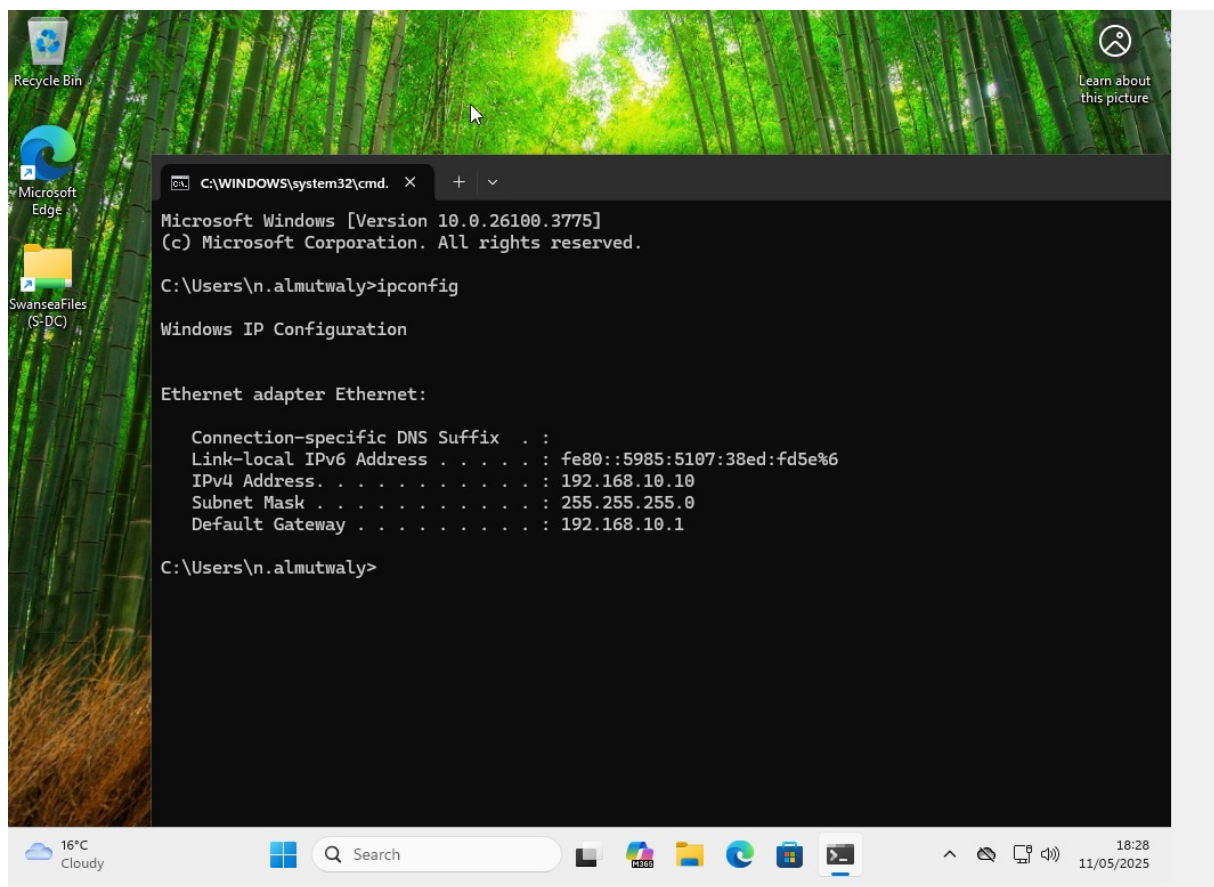
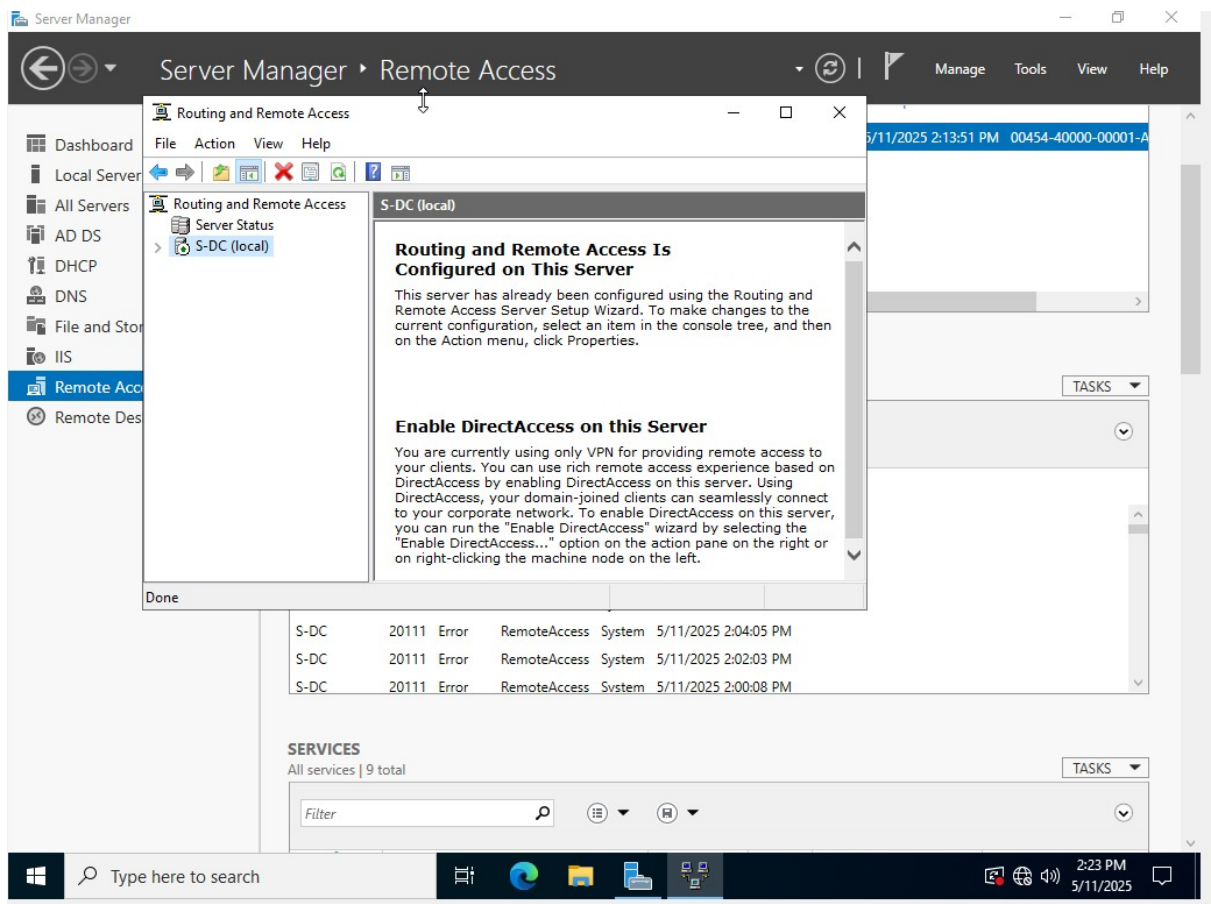


Fig15: DHCP Confirmation

A- Remote access and Services

Then, The Routing and Remote Access was configured to enables secure communication and between Swansea and Cardiff networks through secure VPN connections.



The screenshot below illustrates the illustration of a static route in the Routing and Remote Access Service. It ensures that traffic destined for the 172.22.30.0 is correctly routed through the specified interface.

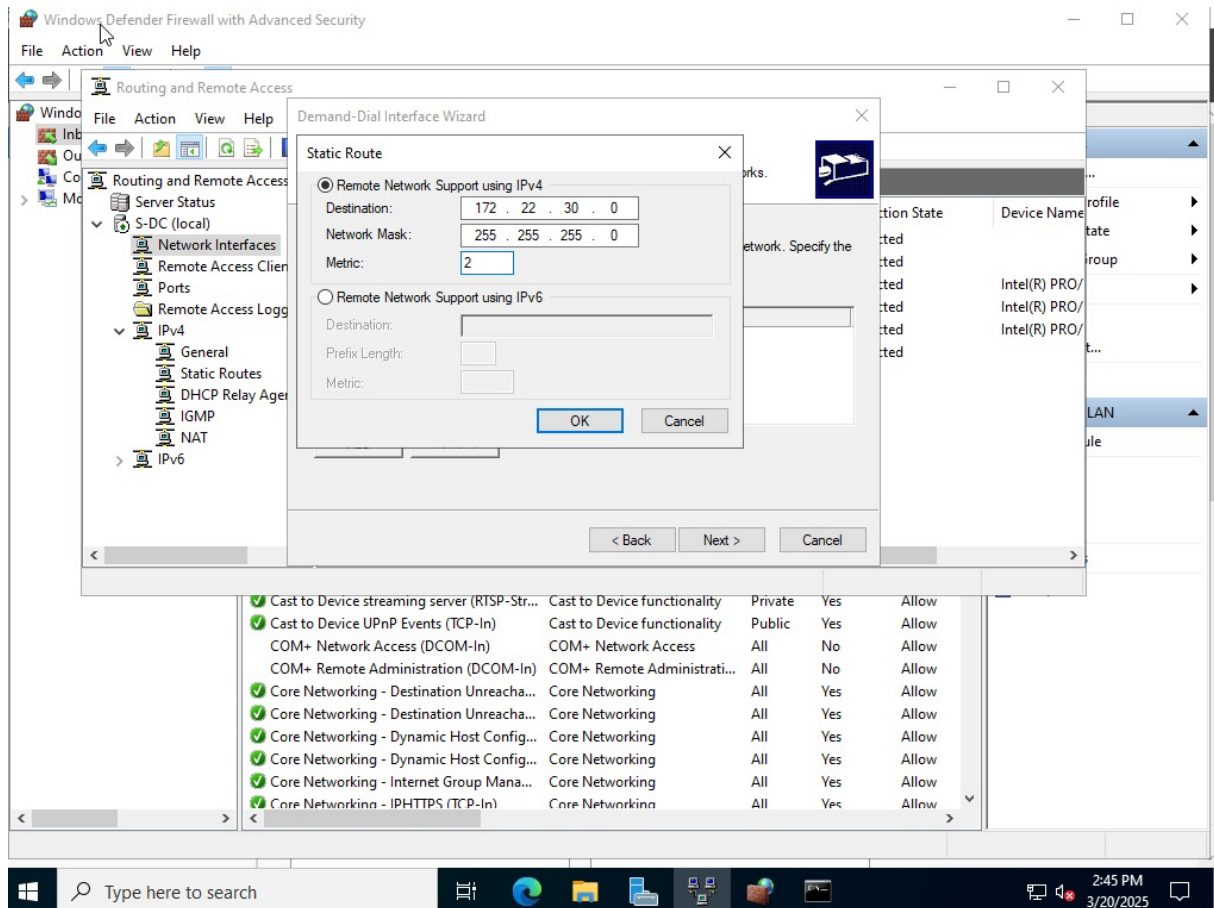


Fig16: Static Route Configuration Swansea to Cardiff in Routing and Remote Access

E- Default Gateway

The Default Gateway was then installed, as it allows devices on a local network to communicate with devices outside that network.

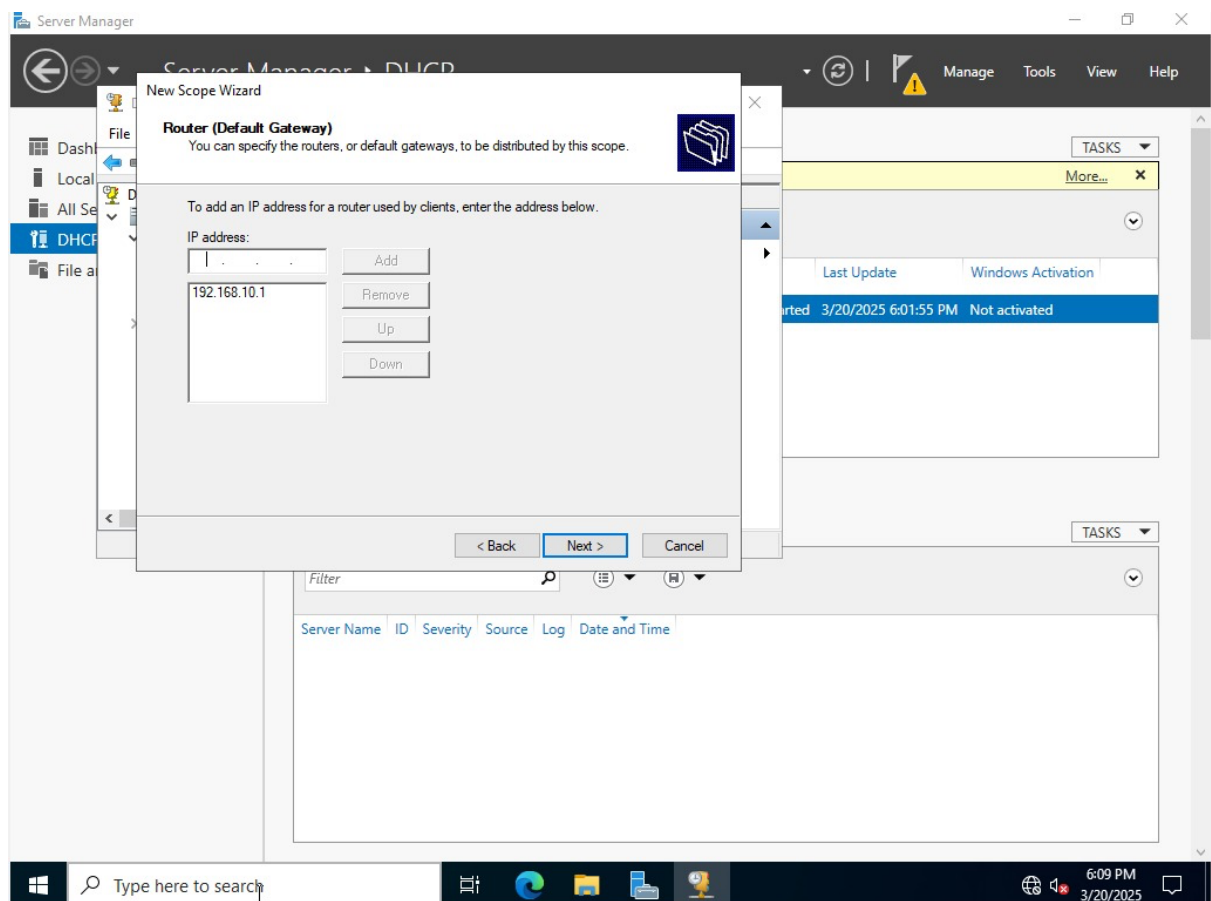


Fig17: Default Gateway

F- VPN

Then we configured the VPN that creates a secure tunnel between the Cardiff and Swansea sites over the Internet.

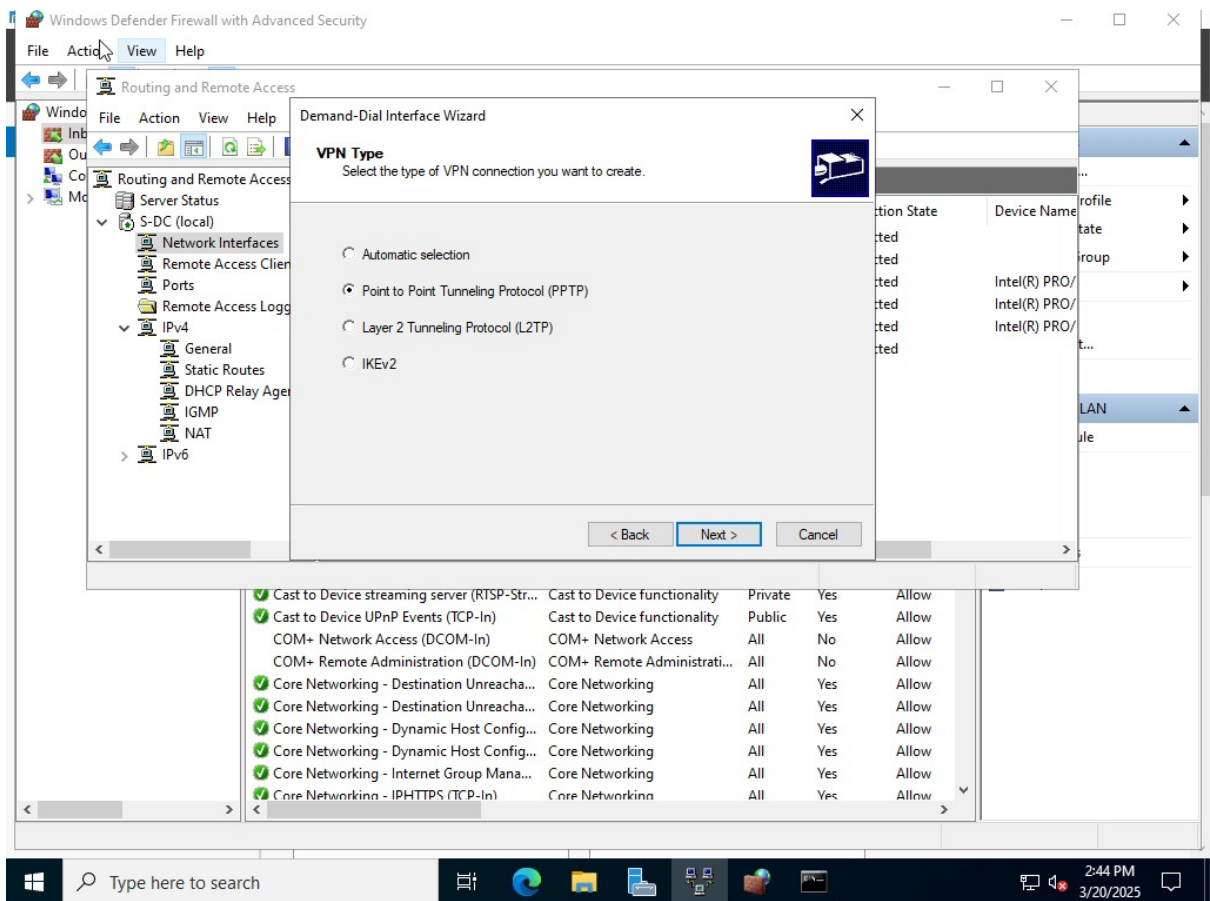


Fig18: VPN Connection Type Configuration

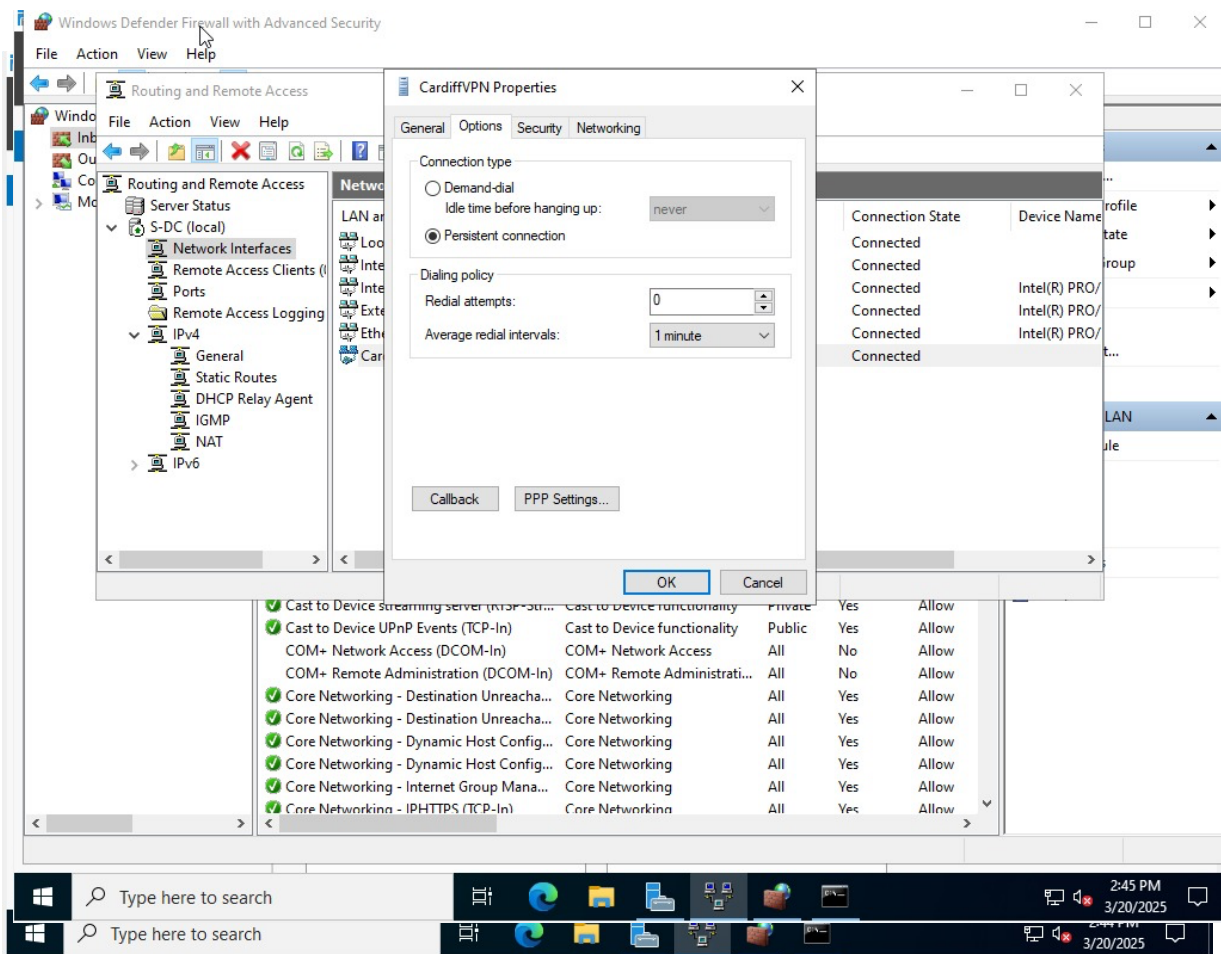


Fig19: VPN Configuration

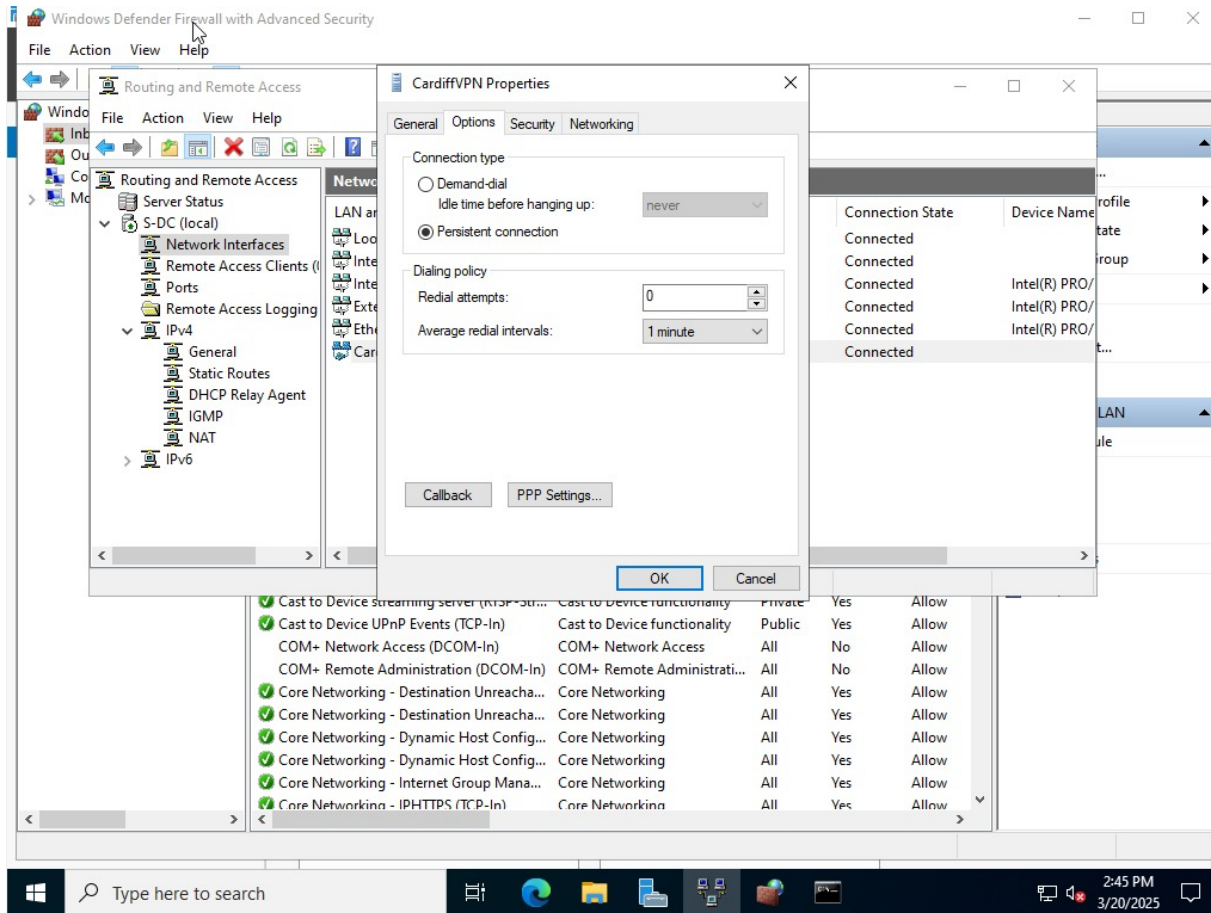


Fig20: Set to Persistent Connexion between Swansea and Cardiff

This screenshot confirms that the VPN has been configured with a constant connection between the Cardiff and Swansea networks.

G- Creation of new user

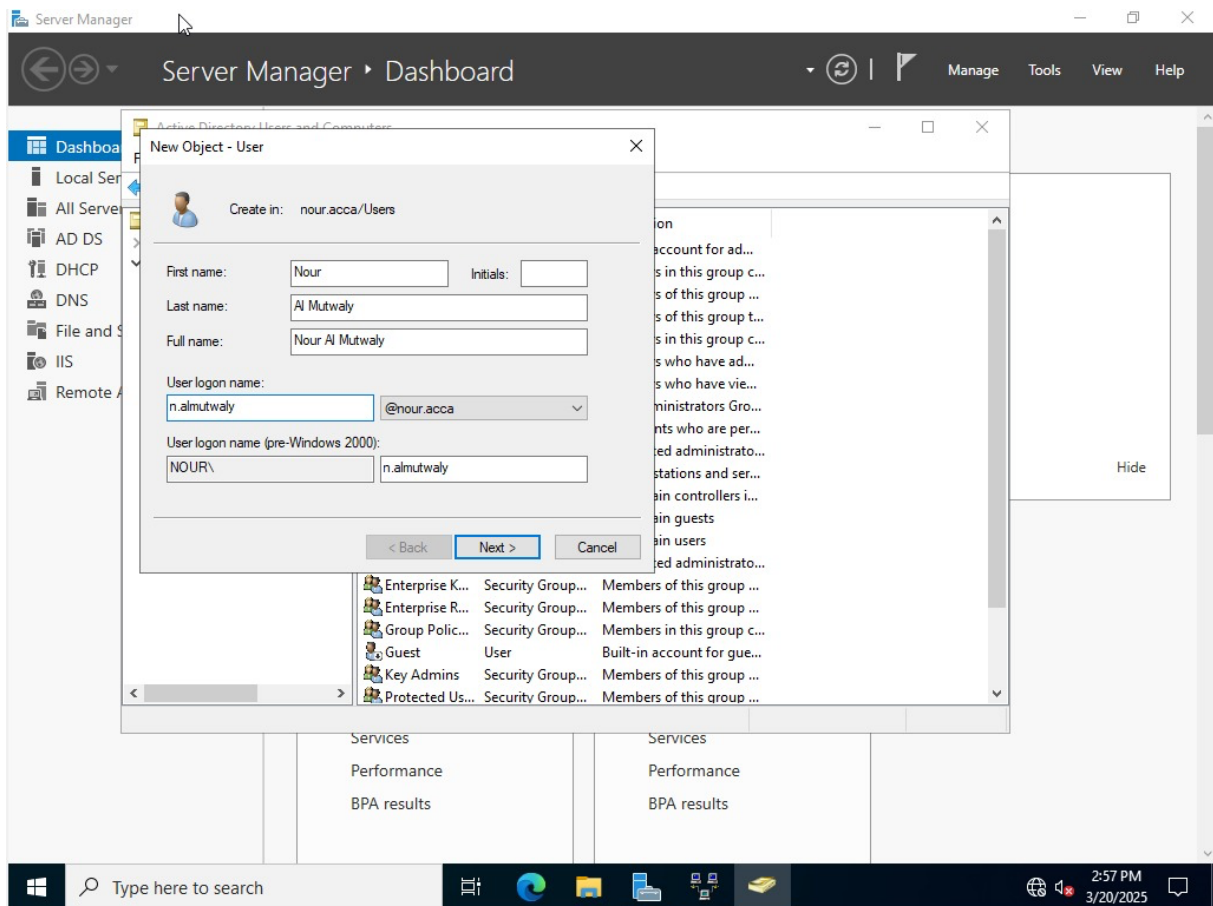


Fig21: Creation of new user

This screenshot shows the creation of a new user account (n.almutwaly) in the nour.acca domain. These accounts will be used for authentication and access control across the Swansea and Cardiff network.

H- Creation of a reverse lookup zone

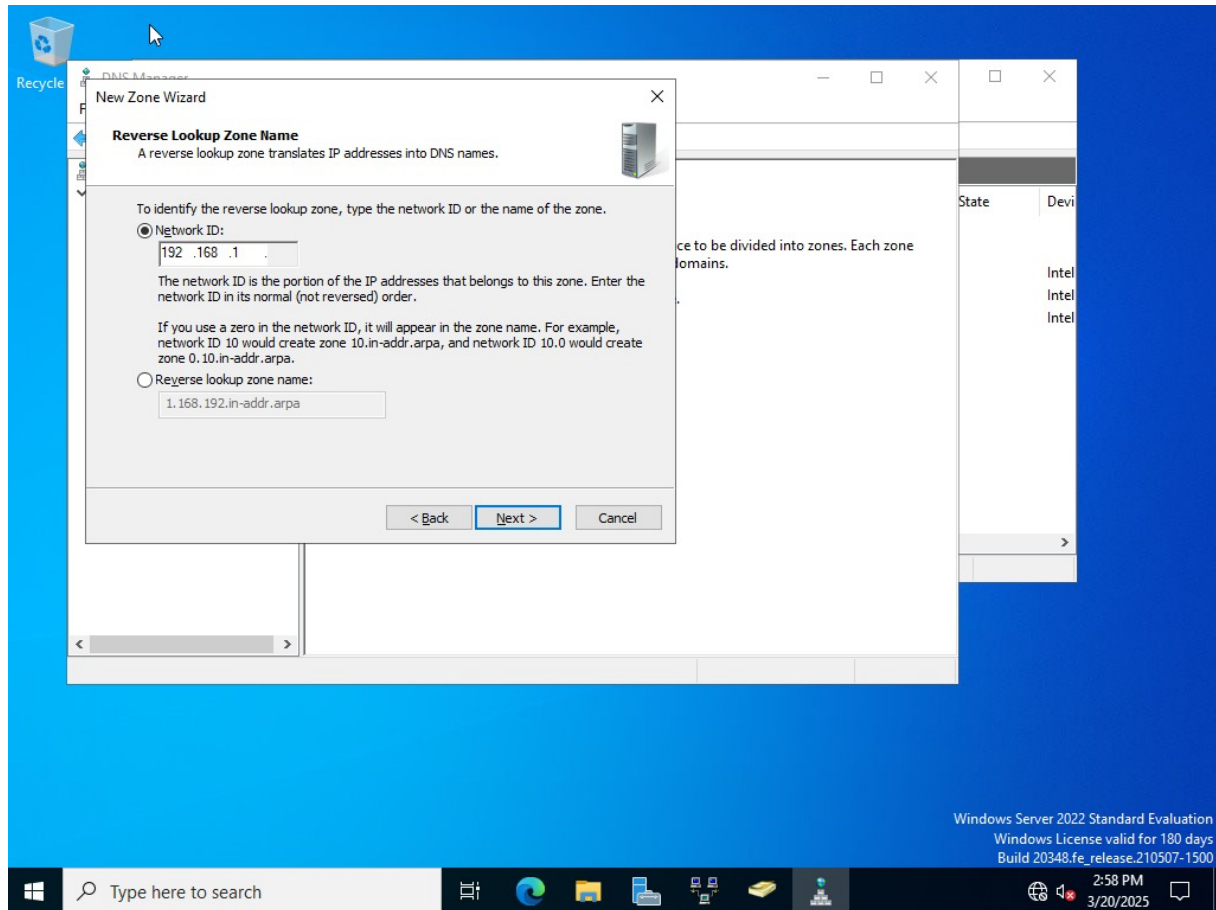


Fig22: Creation of the reverse lookup zone

I- Joining the ClientS1 to domain

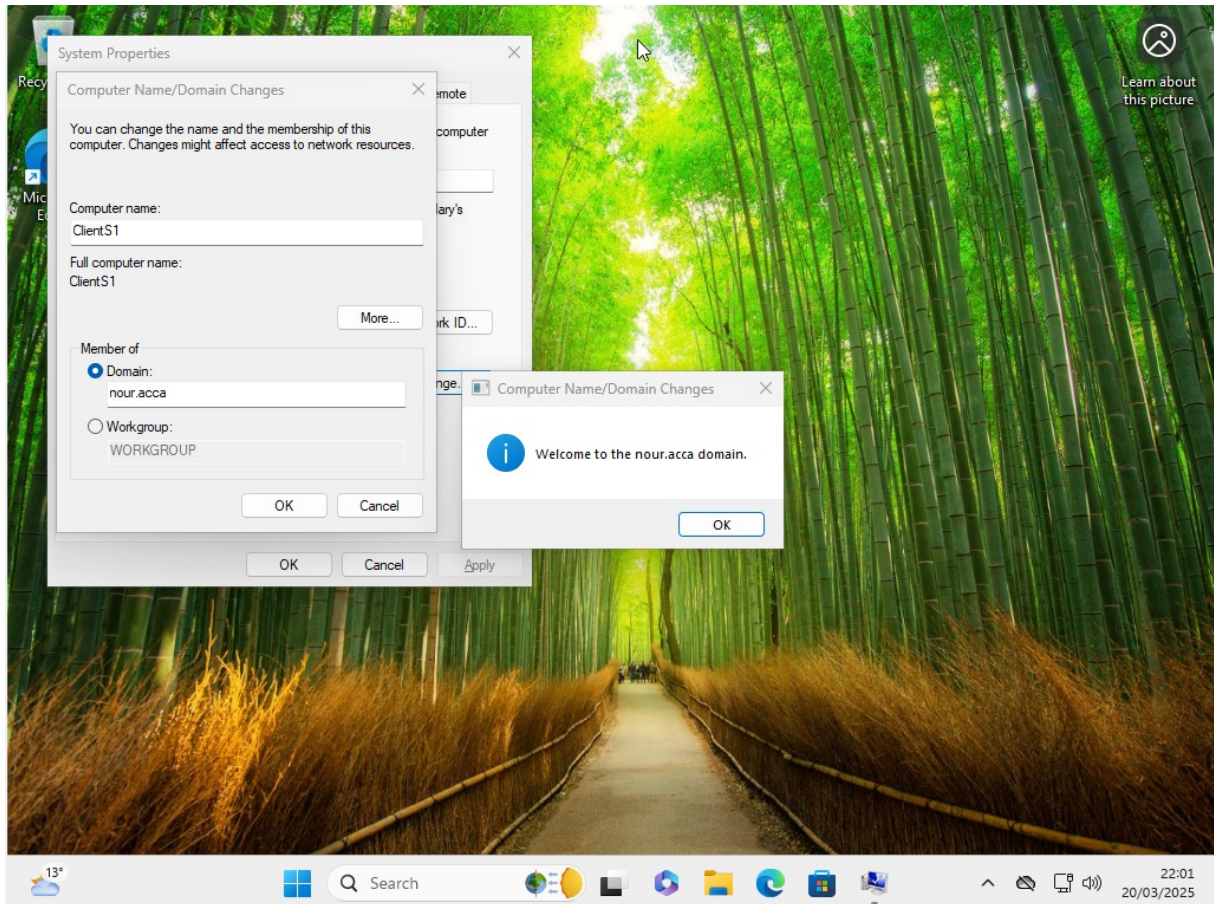


Fig23: ClientS1 Successfully Joined to nour.acca Domain

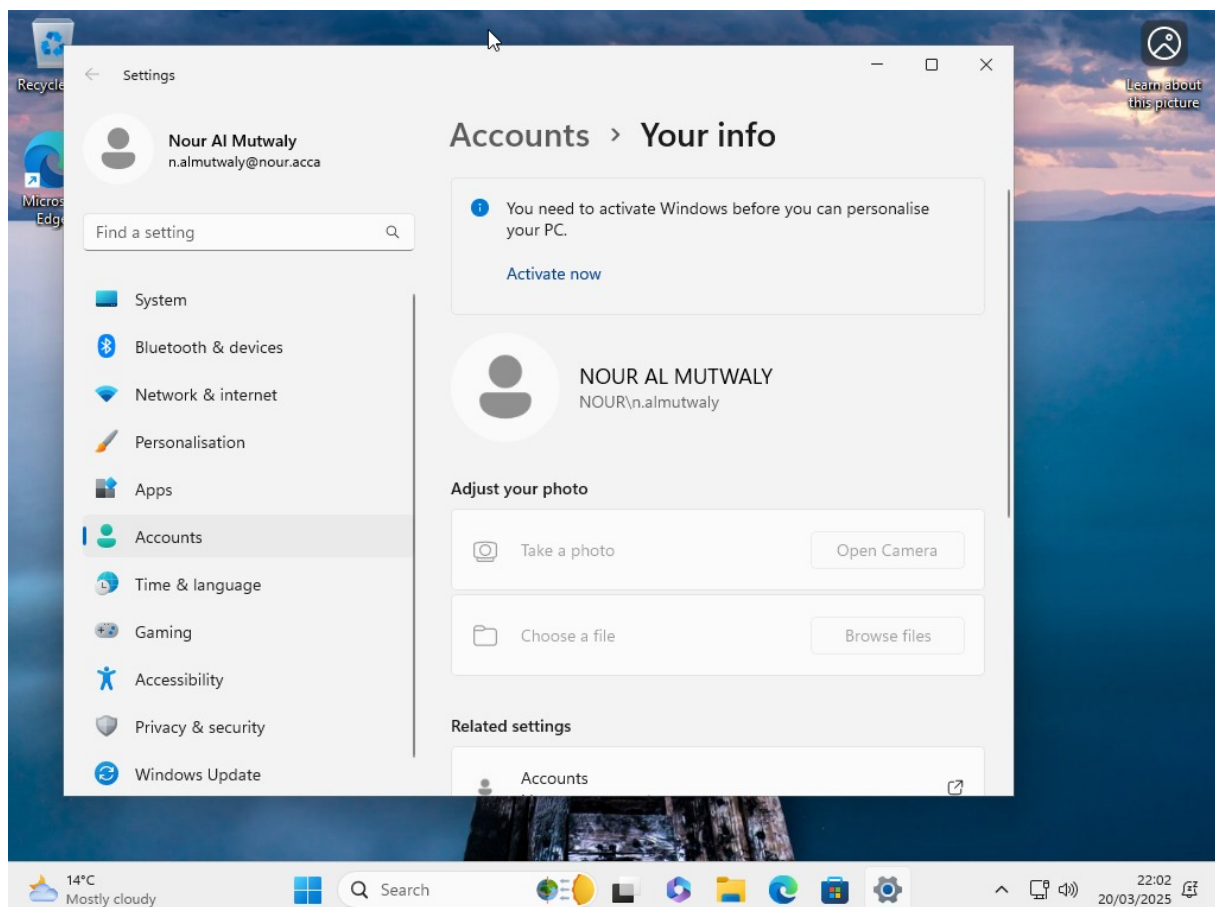


Fig24: Domain User Account Successfully Logged into nour.acca

J- Second Disk Mounted to D:\SwanseaFiles

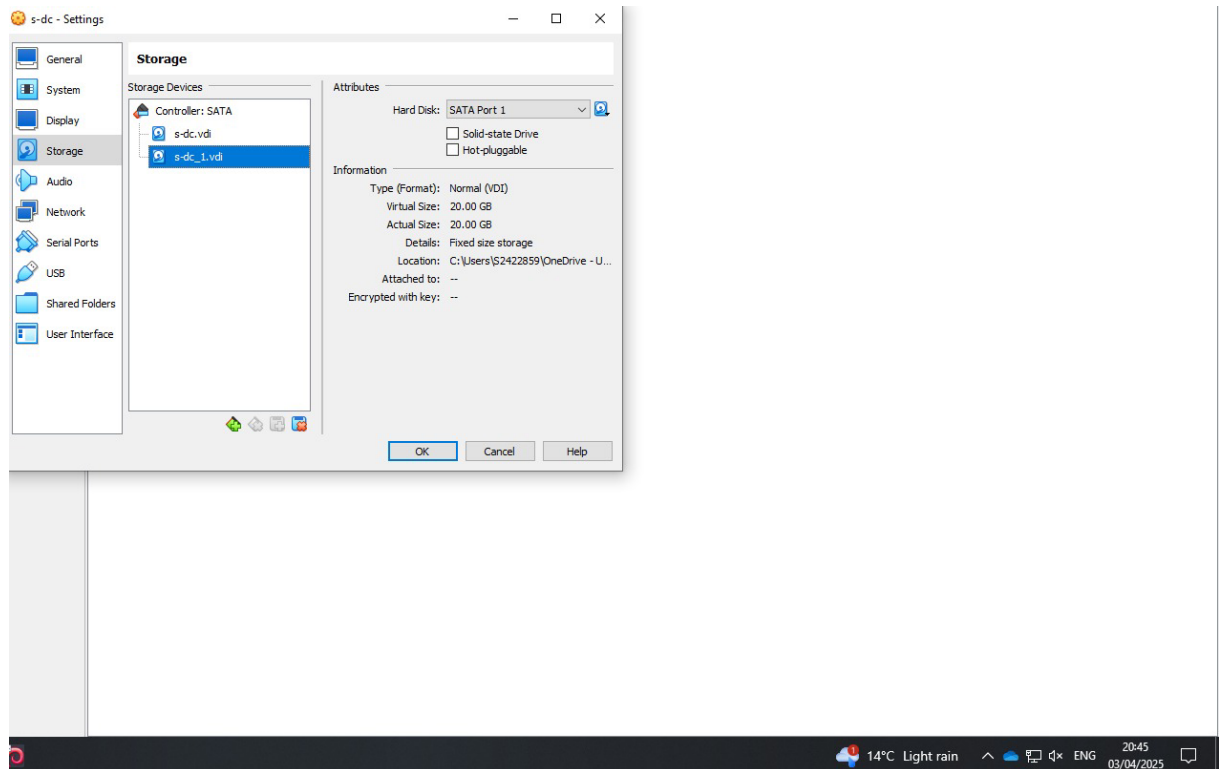


Fig25: Second Virtual Hard Disk

K- Swansea and Cardiff domains

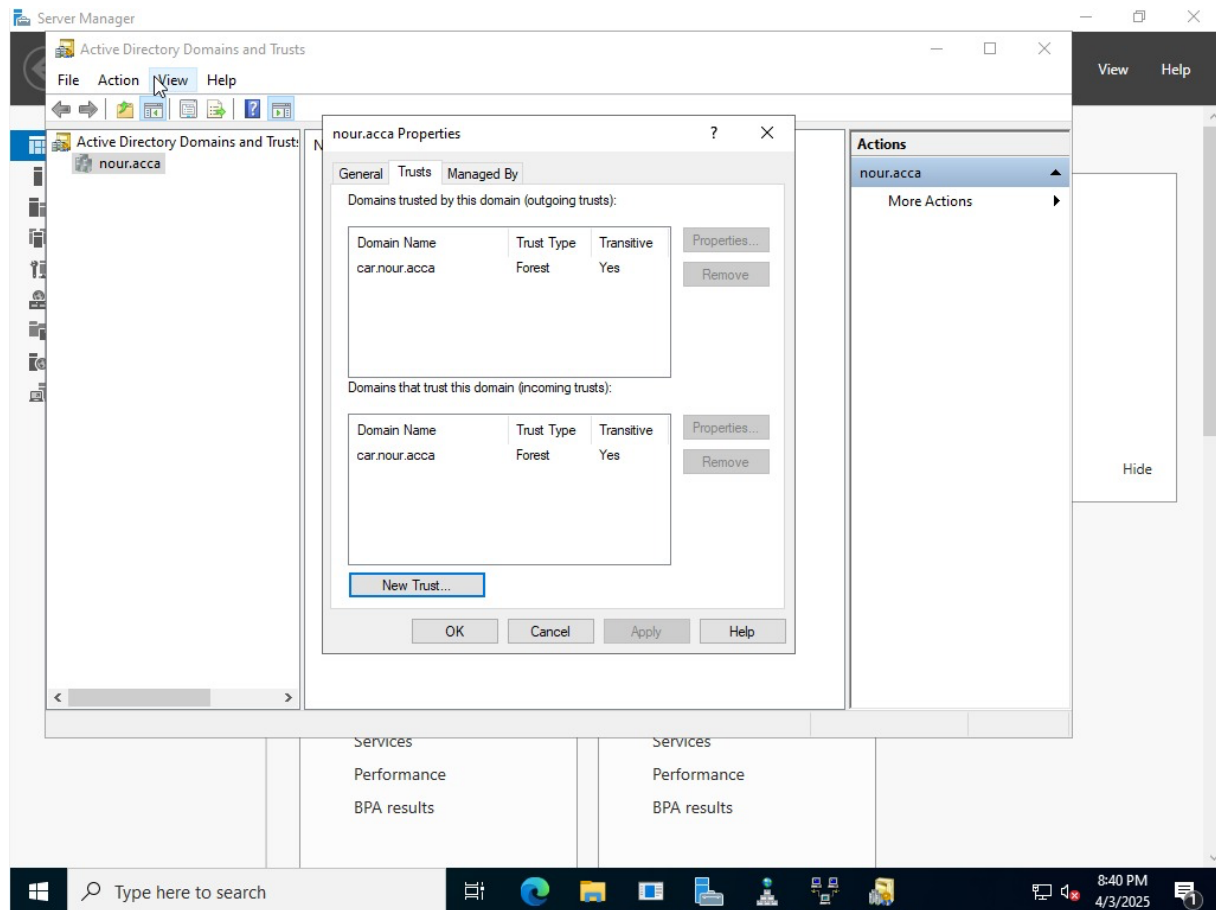


Fig26: Configuration of a trust relationship between Swansea and Cardiff

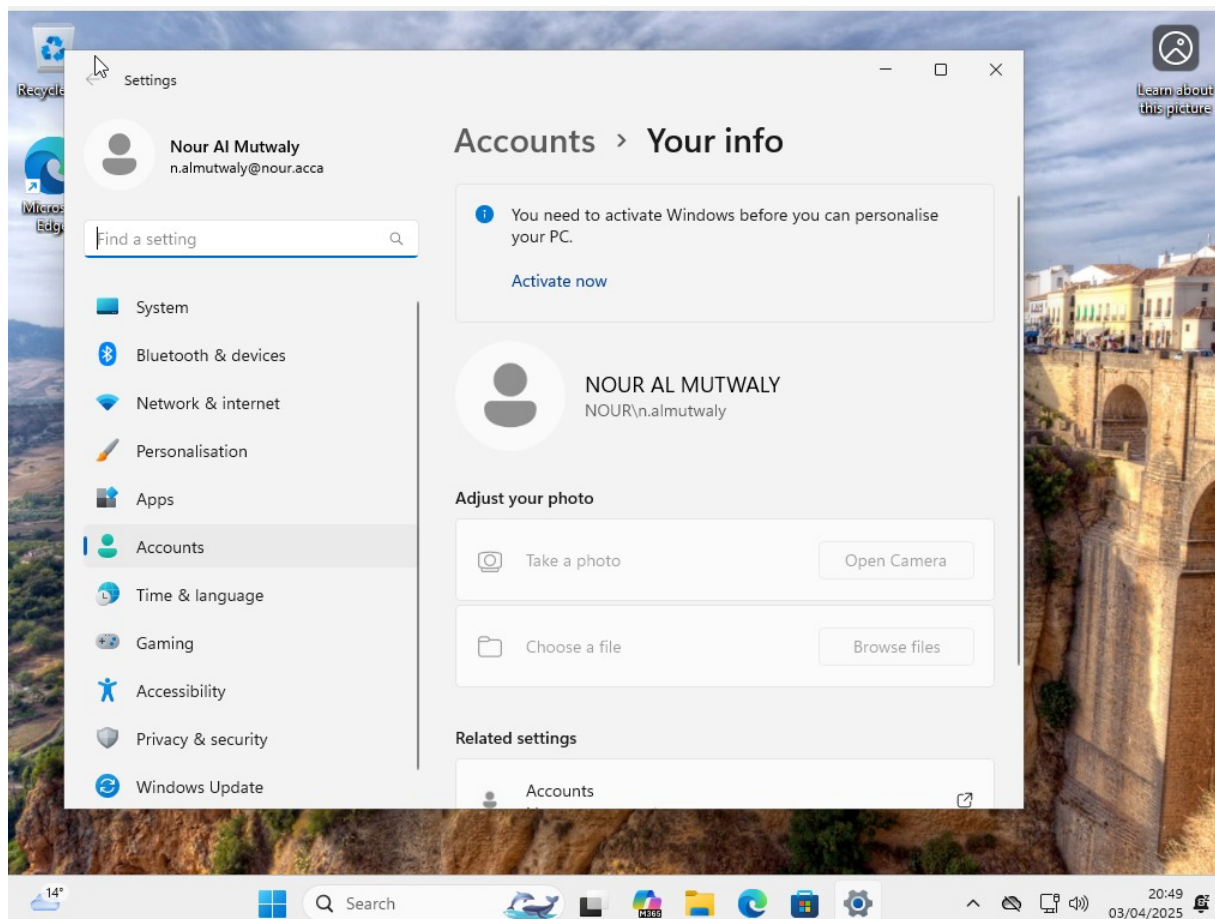


Fig27: Connect Swansea machine to the Client

II- Cardiff Network

A- Routing and Remote Access

```
GNU nano 7.2 /etc/sysctl.conf
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
#net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1

#####
# Additional settings - these settings can improve the network
# security of the host and prevent against some network attacks
# including spoofing attacks and man in the middle attacks through
# redirection. Some network environments, however, require that these
# settings are disabled so review and enable them as needed.
#
# Do not accept ICMP redirects (prevent MITM attacks)
#net.ipv4.conf.all.accept_redirects = 0
#net.ipv6.conf.all.accept_redirects = 0
# _or_
# Accept ICMP redirects only for gateways listed in our default
# gateway list (enabled by default)
# net.ipv4.conf.all.secure_redirects = 1
#
# Do not send ICMP redirects (we are not a router)
#net.ipv4.conf.all.send_redirects = 0
#
# Do not accept IP source route packets (we are not a router)
#net.ipv4.conf.all.accept_source_route = 0
#net.ipv6.conf.all.accept_source_route = 0
#
# Log Martian Packets
#net.ipv4.conf.all.log_martians = 1
#
#####
# Magic system request Key
# 0=disable, 1=enable all, >1 bitmask of sysrq functions
# See https://www.kernel.org/doc/html/latest/admin-guide/sysrq.html
# for what other values do
#kernel.sysrq=438

net.ipv4.ip_forward=1

hour@debian:~$
```

Fig28: Enabling IPv4 Packets Forwarding on Debian

B- DHCP

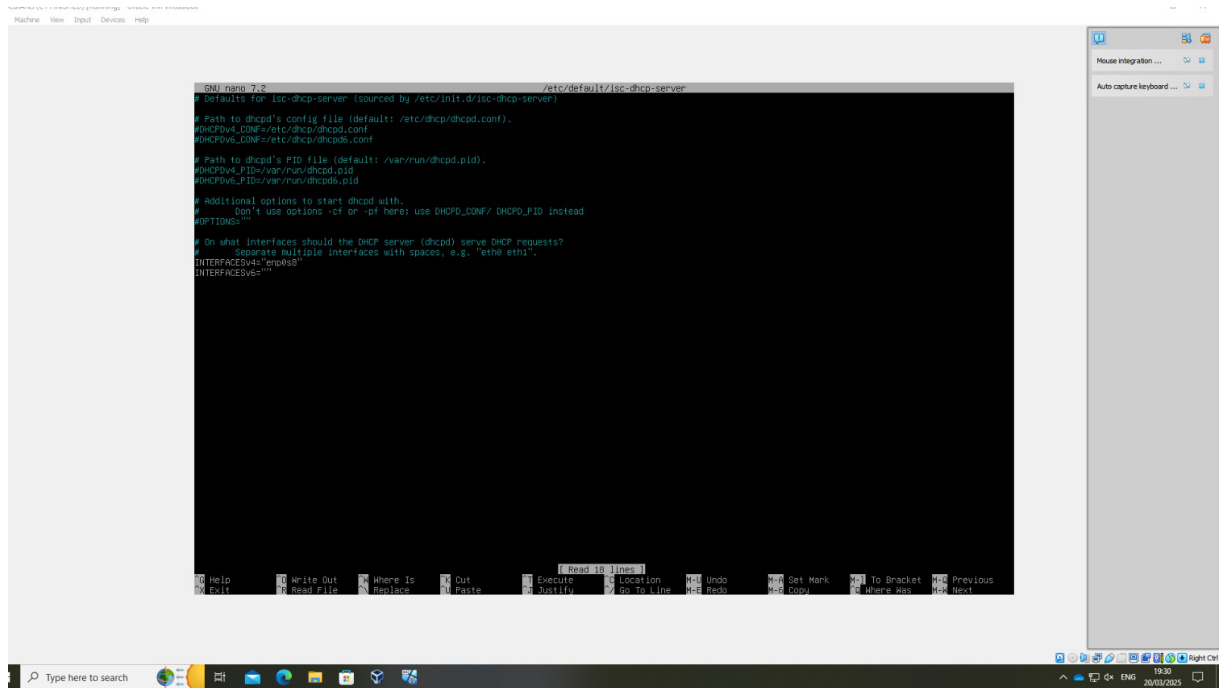


Fig29: Configuring DHCP Server Interface on Debian

```
#!/bin/sh
# should not also be listed as being available for dynamic assignment.
# Hosts for which fixed IP addresses have been specified can boot using
# BOOTP or DHCP. Hosts for which no fixed address is specified can only
# be booted with DHCP, unless there is an address range on the subnet
# to which a BOOTP client is connected which has the dynamic-bootp flag
# set.
#host fantasia [
# hardware ethernet 08:00:07:25:1c:95;
# fixed-address fantasia.example.com;
#]

# You can declare a class of clients and then do address allocation
# based on that. The example below shows a case where all clients
# in a certain class get addresses on the 10.17.224/24 subnet, and all
# other clients get addresses on the 10.0.29/24 subnet.

class "foo" {
# match if substring (option vendor-class-identifier, 0, 4) = "SUNW";
#}

#shared-network 224-29 {
# subnet 10.17.224.0 netmask 255.255.255.0 {
# option routers rtr-224.example.org;
# }
# subnet 10.0.29.0 netmask 255.255.255.0 {
# option routers rtr-29.example.org;
# }
# pool {
# allow members of "foo";
# range 10.17.224.10 10.17.224.250;
# }
# pool {
# deny members of "foo";
# range 10.0.29.10 10.0.29.230;
# }
#}

subnet 172.22.30.0 netmask 255.255.255.0 {
range 172.22.30.10 172.22.30.100;
option routers 172.22.30.1;
option domain-name-servers 8.8.8.8, 1.1.1.1;
option broadcast-address 172.22.30.255;
default-lease-time 600;
max-lease-time 7200;
}
```

Fig30: DHCP Configuration on Debian

C- Creation of the child domain

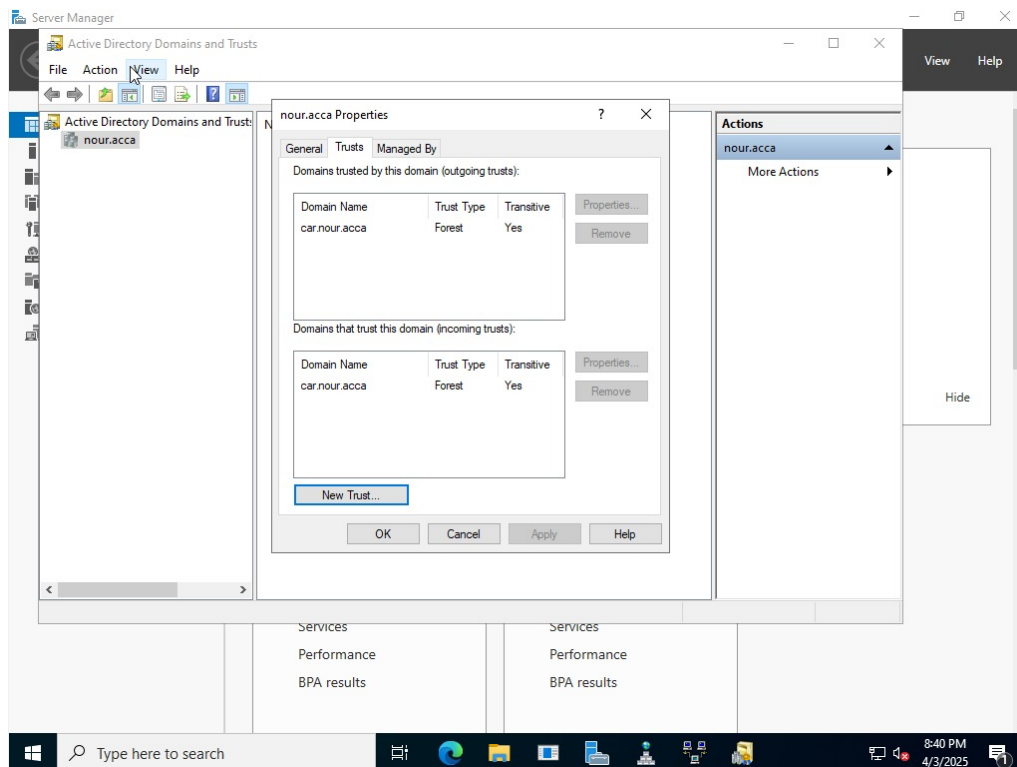


Fig31: Establishing Trust Relationship Between Parent and Child domains in Active Directory

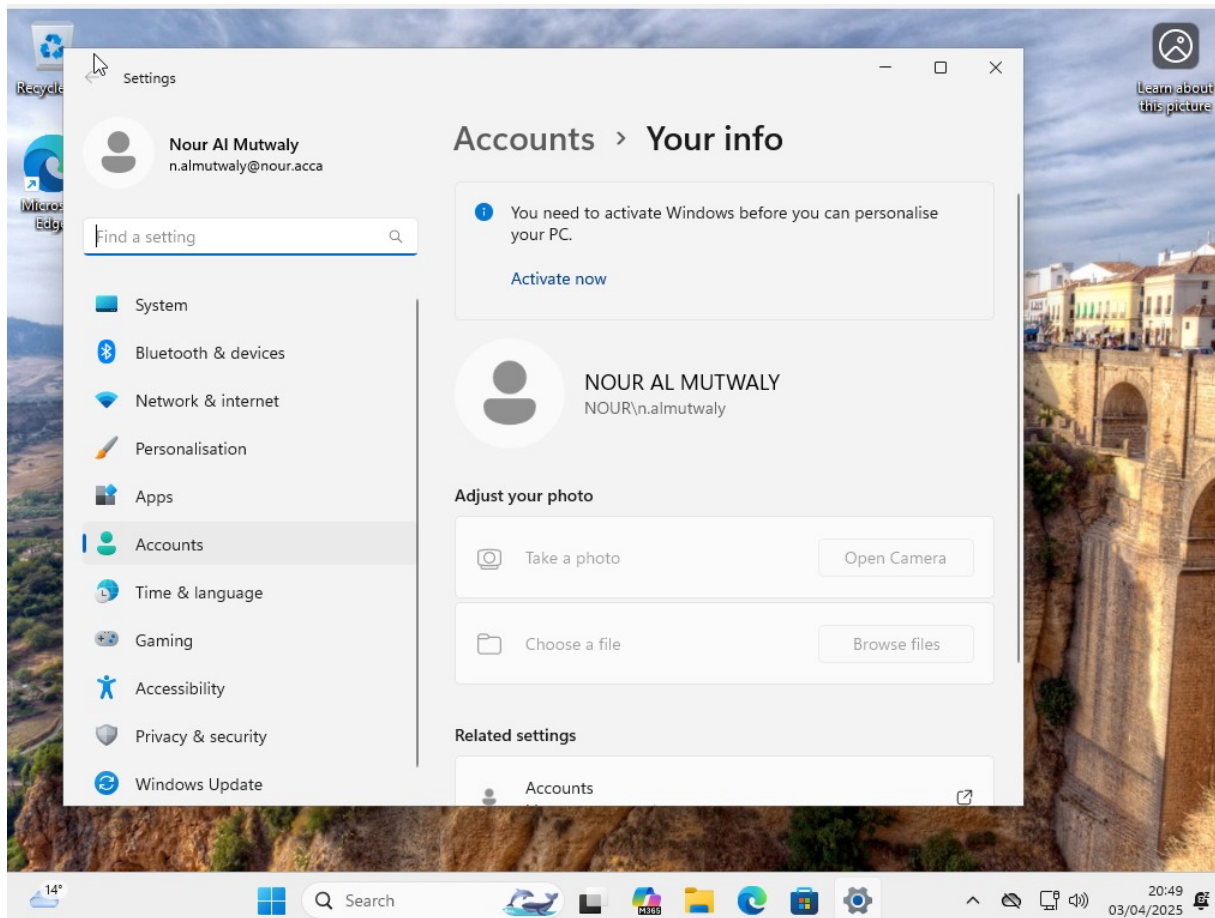


Fig32: Verify Domain User Account Integration on a Client Machine

```
nour@debian:~$ sudo samba-tool domain passwordsettings set --complexity=off
Password complexity deactivated!
All changes applied successfully!
nour@debian:~$ sudo samba-tool domain passwordsettings set --passwordlength=0
Usage: samba-tool domain passwordsettings set <options> [options]

Samba-tool domain passwordsettings set: error: no such option: --passwordlength
nour@debian:~$ sudo samba-tool domain passwordsettings set --min-pwd-length=0
Minimum password length changed!
All changes applied successfully!
nour@debian:~$ samba-tool user create n.almutwaly 'nour' --given-name='Nour' --surname='Al Mutwaly'
ldb: tdb(/var/lib/samba/private/sam.ldb): tdb_open_ex: could not open file /var/lib/samba/private/sam.ldb: Permission denied
Unable to open tdb '/var/lib/samba/private/sam.ldb': Permission denied
Failed to connect to 'tdb:///var/lib/samba/private/sam.ldb' with backend 'tdb': Unable to open tdb '/var/lib/samba/private/sam.ldb': Permission denied
ERROR(ldb): Failed to add user 'n.almutwaly': - Unable to open tdb '/var/lib/samba/private/sam.ldb': Permission denied
nour@debian:~$ sudo samba-tool user create n.almutwaly 'nour' --given-name='Nour' --surname='Al Mutwaly'
User 'n.almutwaly' added successfully
nour@debian:~$ ate
-bash: ate: command not found
nour@debian:~$ date
Thu 3 Apr 20:19:54 BST 2025
nour@debian:~$
```

Fig33: Configure Samba AD Domain Controller Password Policies and Create Domain User

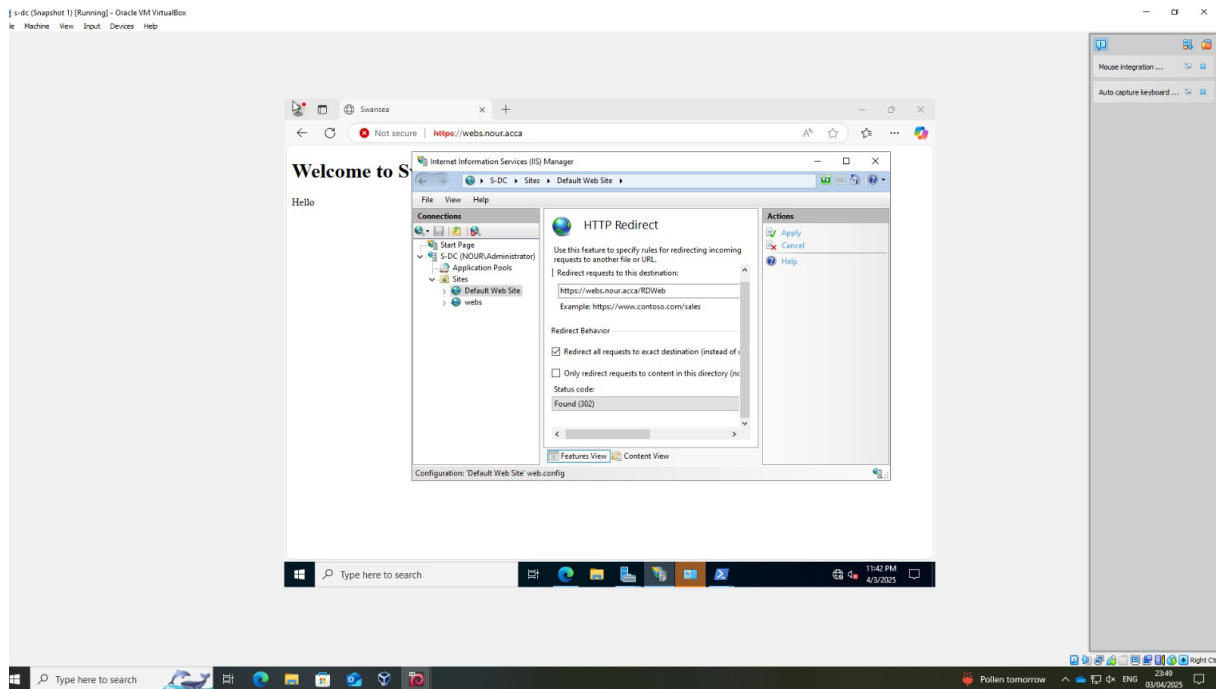


Fig34: Configuring IIS to Redirect Website

D- Shared Folder

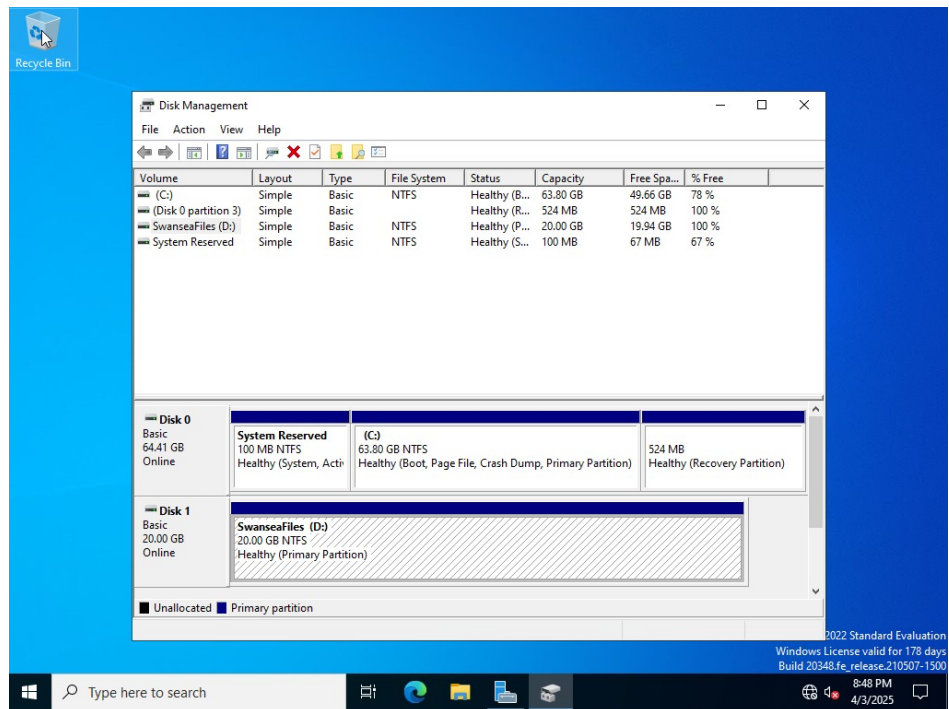


Fig35: Disk Configuration for File Sharing

E- Website Creation

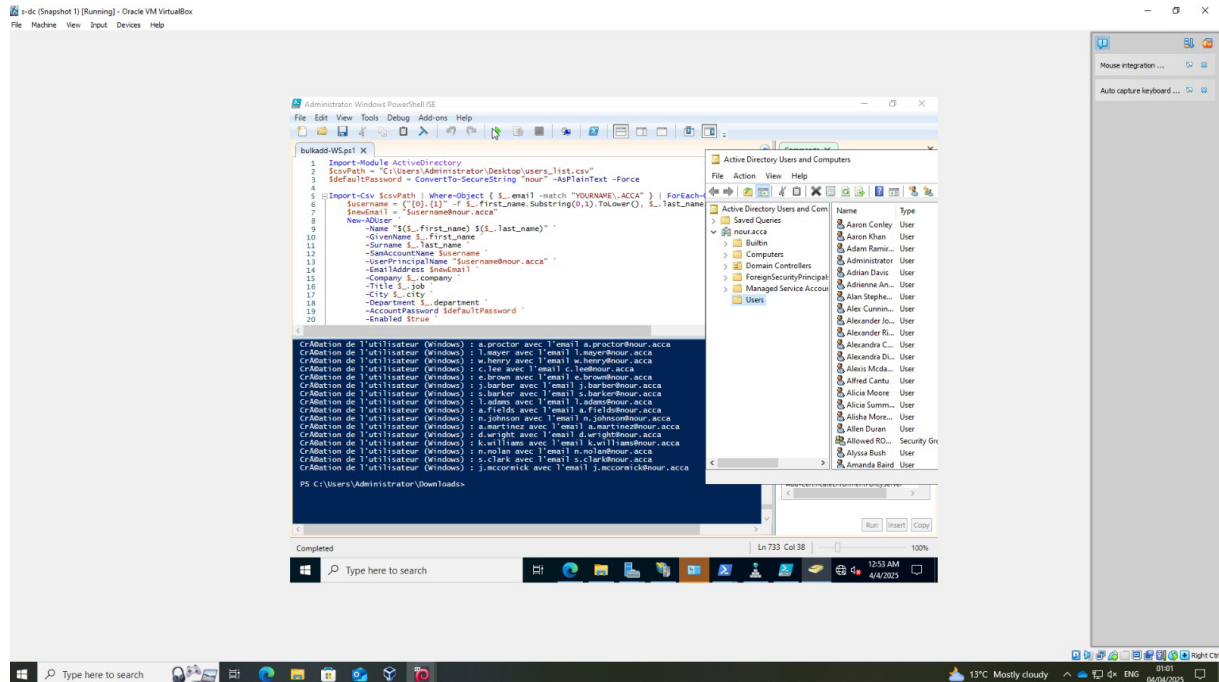


Fig36: Website Creation

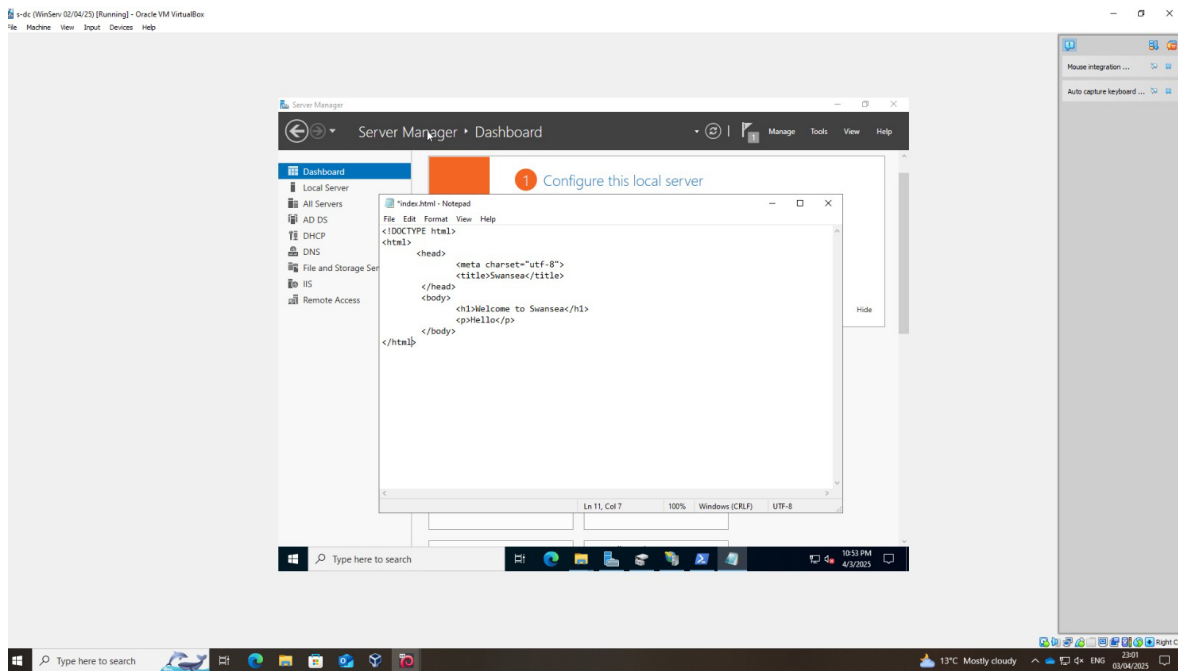


Fig37: Creating a Local Homepage on the Swansea IIS Web Server

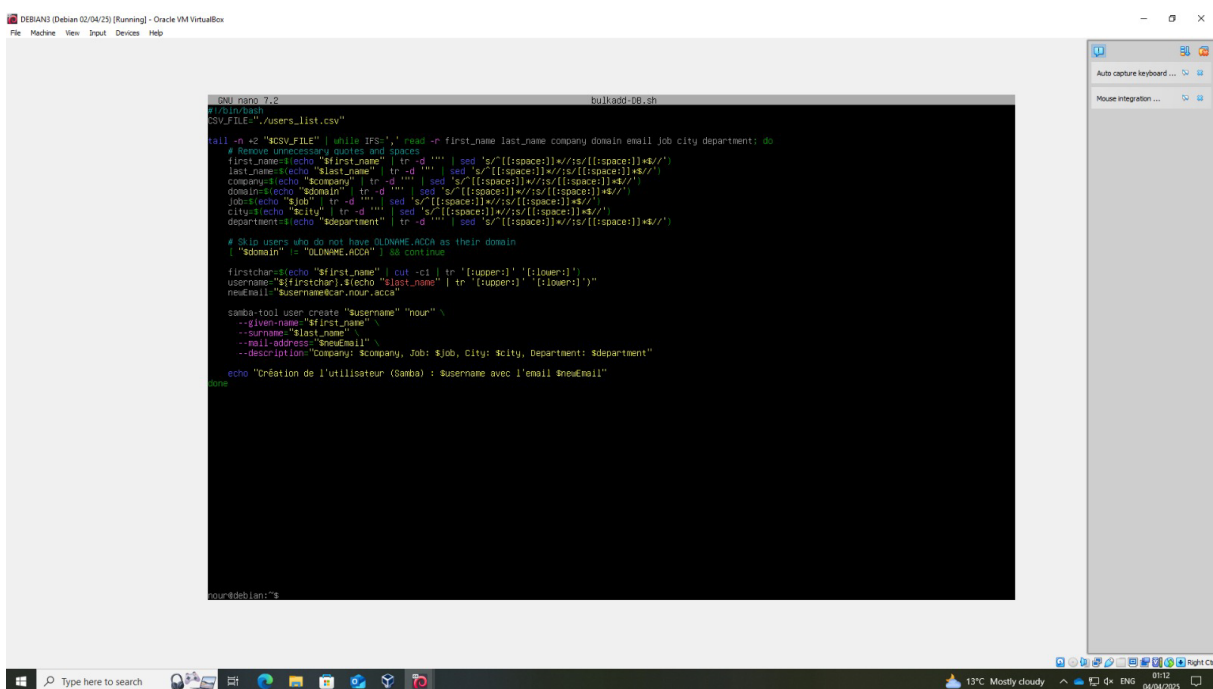


Fig38: Automatic User Creation on Debian Server