

Report of the Network Operating Systems and Services Practice

Introduction

This report demonstrates the deployment and interconnection configuration process for two different networks in a virtualised environment. This work is a continuation of coursework1, and the goal is to achieve a cross-network site connection and service configuration, including domain controller settings, site-to-site VPN, DHCP service, and other content. This project was completed through a collaborative effort between Windows and Debian systems, resulting in the construction of an interconnected network environment. At the same time, in order to facilitate the reading, I put all the photos in the appendix, and give the corresponding number to facilitate the search.

My practice and results

In the whole process of network architecture and implementation, I set coursework1 configured winserver2022 and win10 as Swansea network, and debian and ubuntu as Cardiff network. I build the Swansea network and Cardiff network separately.

The first step in the setup of the Swansea network was the installation of the Routing and Remote Access Service (RRAS) (Figure 1.1.1). The activation of this service is a prerequisite to enable subsequent VPN tunnel establishment. Once the configuration was complete, I added new rules to the Windows Firewall via code to allow ICMPv4 and ICMPv6 inbound and outbound traffic, thus ensuring that connectivity between hosts could be quickly verified via the Ping command during the testing phase.

I then configured the Site-to-Site VPN connection. The Demand-Dial interface was created through the RRAS console, the PPTP protocol was attempted to be selected, and the Interface was created with the connection type set to 'Persistent' (Figure 1.2).

After the VPN connection was established, I proceeded to create a new Active Directory forest and domain in Windows Server, named yuelianghan.acca (Figure 1.3.1). This is the core of the entire network authentication system. Configuring the domain controller not only supports centralised account management, but also facilitates uniform privilege assignment. Afterwards, I checked whether the forest and the domain were successfully configured through PowerShell (Figure 1.3.2), which facilitates the subsequent deployment of DNS and user system integration.

In the DNS configuration, I added forward A records (Figure 1.4) and reverse PTR records (Figure 1.5) to the address ranges of 192.168.1.0/24 and 192.168.2.0/24, respectively. A records are used for resolving hostnames to IP addresses, while PTR records are used for reverse resolution of hostnames. In many authentication scenarios, especially in certain authentication or logging systems, reverse resolution is a basic requirement for stable system operation. In order to ensure

accuracy, I meticulously checked the correspondence between the name and address of each host during configuration, and conducted forward and reverse resolution tests after configuration.

After that, I created domain user accounts in bulk through PowerShell scripts based on the `users_list_updated.xlsx` file provided by my teacher (Figure 1.6.1). Using a script not only improves efficiency, but also avoids errors that may occur in manual operations. I've set up automatic generation of SAM account names (using the format 'First letter. Surname' format), and specified the default password, email address, and account number for each user, and then checked the list of the created account information through PowerShell to confirm the accuracy of the code at the end (Figure 1.6.2).

In terms of hard disk configuration, considering that the system already has a D drive, I configured the second mount disk to be an E drive (Figure 1.7.1) and created a folder named 'SwanseaFiles' for future file sharing. During the mounting process, I initialized, partitioned and formatted the disc using Disk Management (NTFS format), then manually changed the disc letter and set the sharing parameters and access controls (Figure 1.7.2).

On the Swansea network, I configured the address from 192.168.1.10 to 192.168.1.20, setting options such as the default lease time, default gateway with DNS server addresses (Figure 1.8). With this setup, clients in the network can automatically obtain the required IP and related configuration when they join the network, avoiding manual configuration again.

Once the configuration was complete, I joined ClientS1, which is win10, to the `yuelianghan.acca` domain. During this process, to ensure that the domain controller can be discovered, I first set the DNS of this client to the domain controller IP (Figure 1.9.1). The domain was then joined using the 'Change Settings' option in System Properties (Figure 1.9.2). Once joined, I logged on using one of the test user accounts created earlier (Figure 1.10) to verify that the domain authentication was working.

After completing the deployment of the Swansea network, I turned my focus to the configuration of the Cardiff network. This network uses the Debian operating system, which is quite different from Windows Server in terms of architecture and tools. The configuration process therefore required a suitably adapted approach to ensure smooth collaboration between the two systems.

Firstly, I configured firewall rules in Debian via iptables to allow ICMP traffic and enabled IPv4 packet forwarding to support VPNs and internal routing (as shown in Figures 2.1.1 and 2.1.2). Meanwhile, when I tried to install and configure PPTP in Debian (Figure 2.2.1), I found that I could not effectively penetrate the network to

establish a tunnel due to the NAT and Host-only network restrictions in the VM settings. So after discussing and trying with my classmates, I switched to WireGuard protocol, which is lightweight, efficient, encrypted and secure, and better adapted to cross-platform virtual environments. I reconfigured the corresponding public and private keys on the Windows Server and Debian server respectively (as shown in Figure 2.2.2), and set up a fixed IP segment and port mapping to ensure stable communication between the two sides. Therefore, after debugging the VM setting and network setting, I finally achieved connection between the two servers.

Considering that Debian itself doesn't have the capability of a Windows domain controller, I didn't create a subdomain `car.yuelianghan.acca` in Cardiff's network, but chose to join its devices directly to the main domain. During the join process, I used the Samba and Realm tools to configure the Debian client for domain joining (as shown in Figures 2.3.1 and 2.3.2) to ensure that it can interact with the AD domain properly.

The DHCP service was similarly deployed in the Cardiff network. I chose to use the `isc-dhcp-server` tools and defined the address from `192.168.56.100` to `192.168.56.110` by editing the `/etc/dhcp/dhcpd.conf` file (Figure 2.4). In addition to the basic address range settings, I configured the router, broadcast address, and DNS server options to ensure that the network parameters were automatically configured completely.

Once configured, I connected ClientC1 to the network to ensure that it obtained its IP from DHCP and was able to successfully resolve domain names. To test network connectivity, I attempted to Ping the servers in the Swansea network from ClientC1, and domain controller login interfaces (Figure 2.5).

At this point, the Swansea and Cardiff networks were successfully interconnected in the virtual environment, and could ping each other (Figure 3).

Following this, I began experimenting with creating shared disks by setting the `E:\SwanseaFiles` folder I had added earlier as a shared directory and configuring the access permissions so that specific users within the domain, such as those defined in `users_list.xlsx`, had full control (Figure 4.1), while other network users have read-only access (Figure 4.2). A `New Text Document.txt` file was added to it (Figure 4.3).

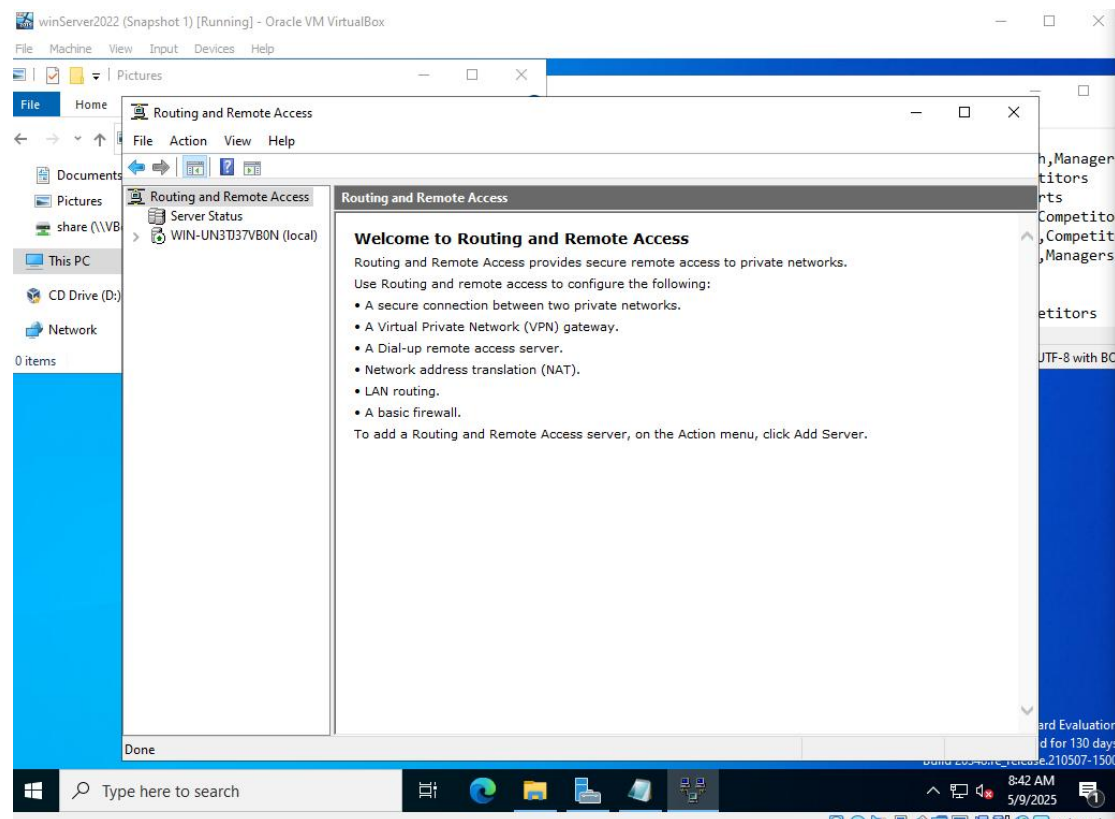
I then tried to find the folder in the Debian client on the Cardiff network, but probably because I was not finding the name of the corresponding system, I was not successful in finding the `SwanseaFiles` folder. To simulate the goal of this task, I used the Guest Additions provided by VirtualBox. By creating the `SwanseaFile` folder on the host and sharing it, I managed to see the `/media/sf_SwanseaFiles`

folder appearing on debian and read the New Text Document.txt file inside it, whose content was consistent with the original file (Figure 4.4).

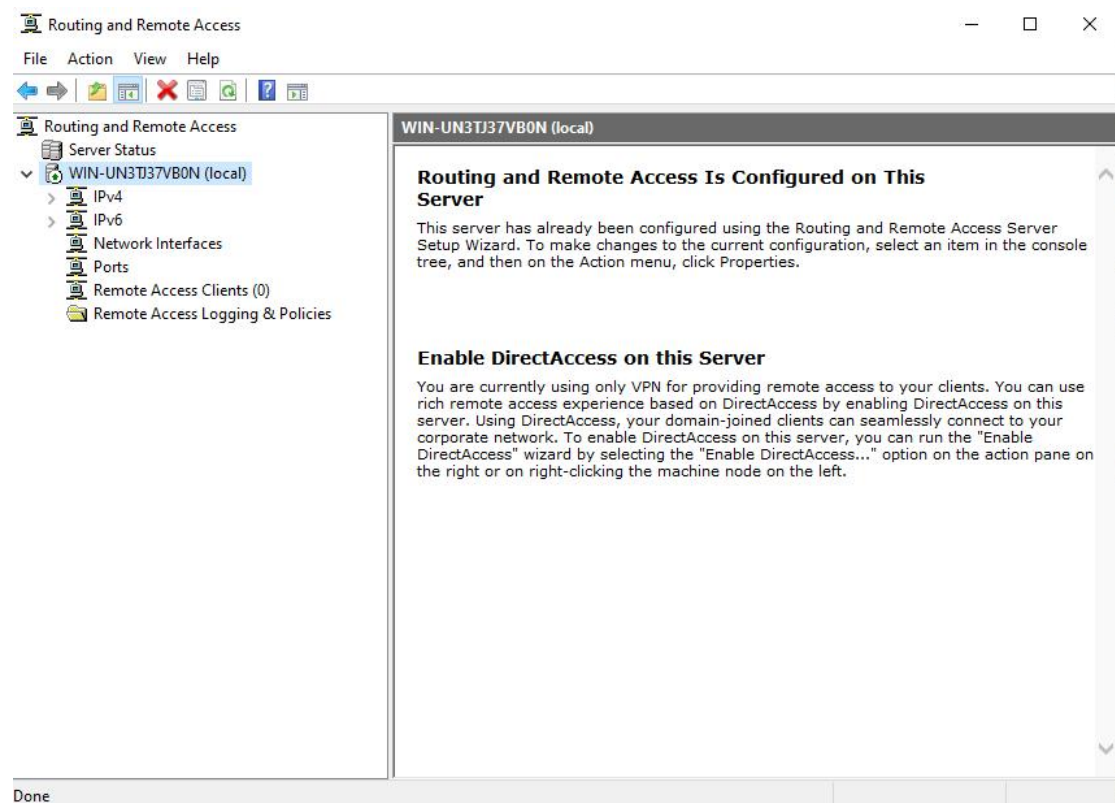
Reflection and Evaluation

During the implementation of this project, several real-world problems were encountered, such as the limitations of virtualised network environments on PPTP VPNs, and the compatibility between the linux systems and Windows domain controllers. By using a more modern VPN protocol, WireGuard, and reasonably adjusting the architecture and permission settings, cross-platform interconnection was achieved. The practice enhanced my understanding of network services, identity management, DNS/DHCP configuration, security privilege assignment, etc., and I learnt to flexibly cope with the challenges brought by operating system differences.

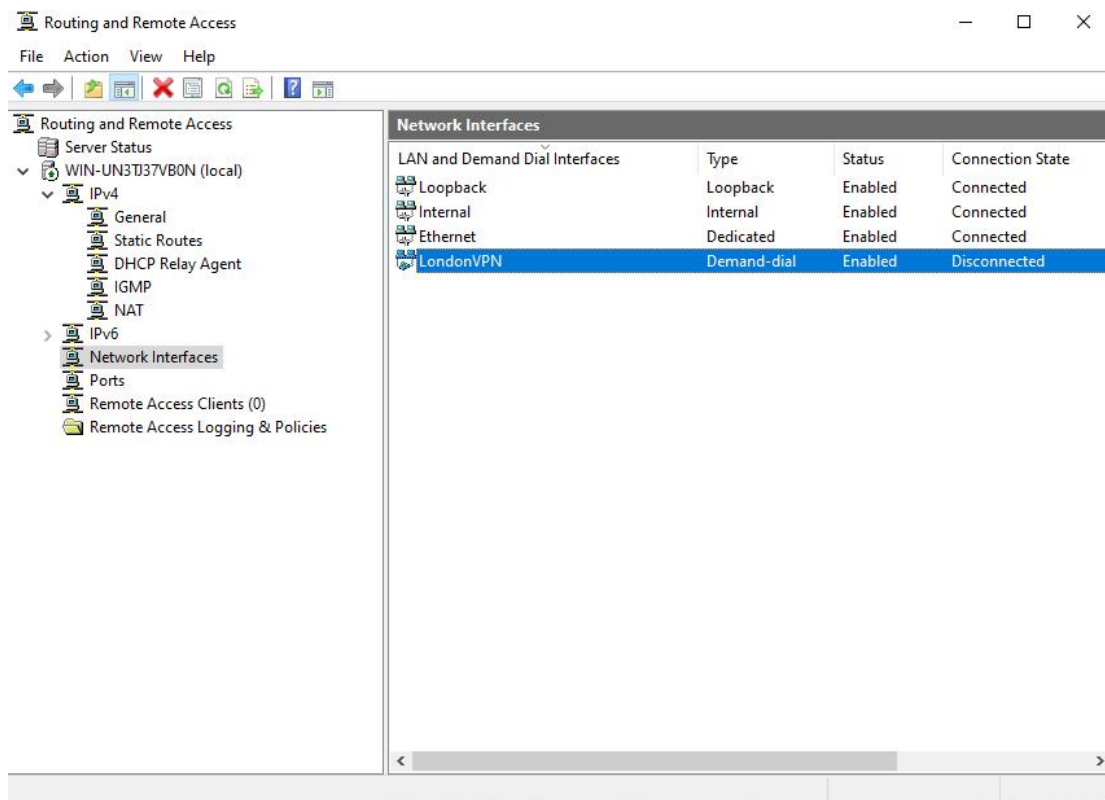
Appendix



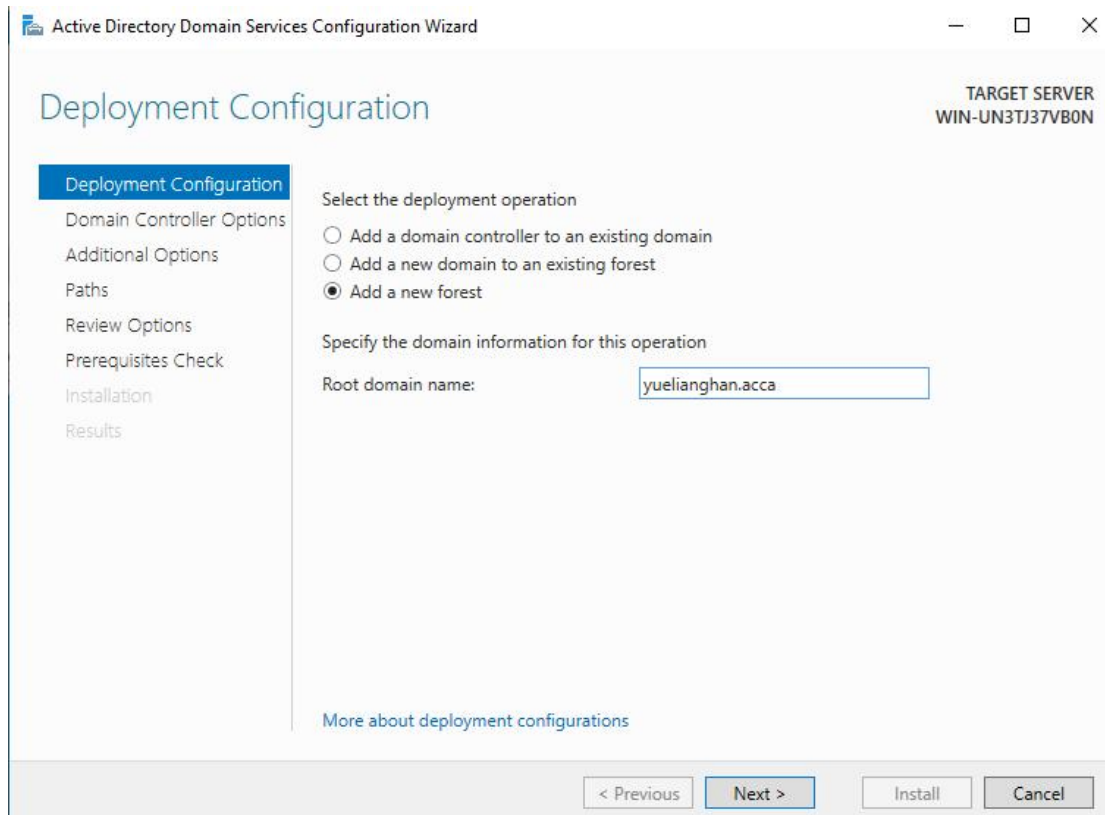
1.1.1



1.1.2



1.2



1.3.1

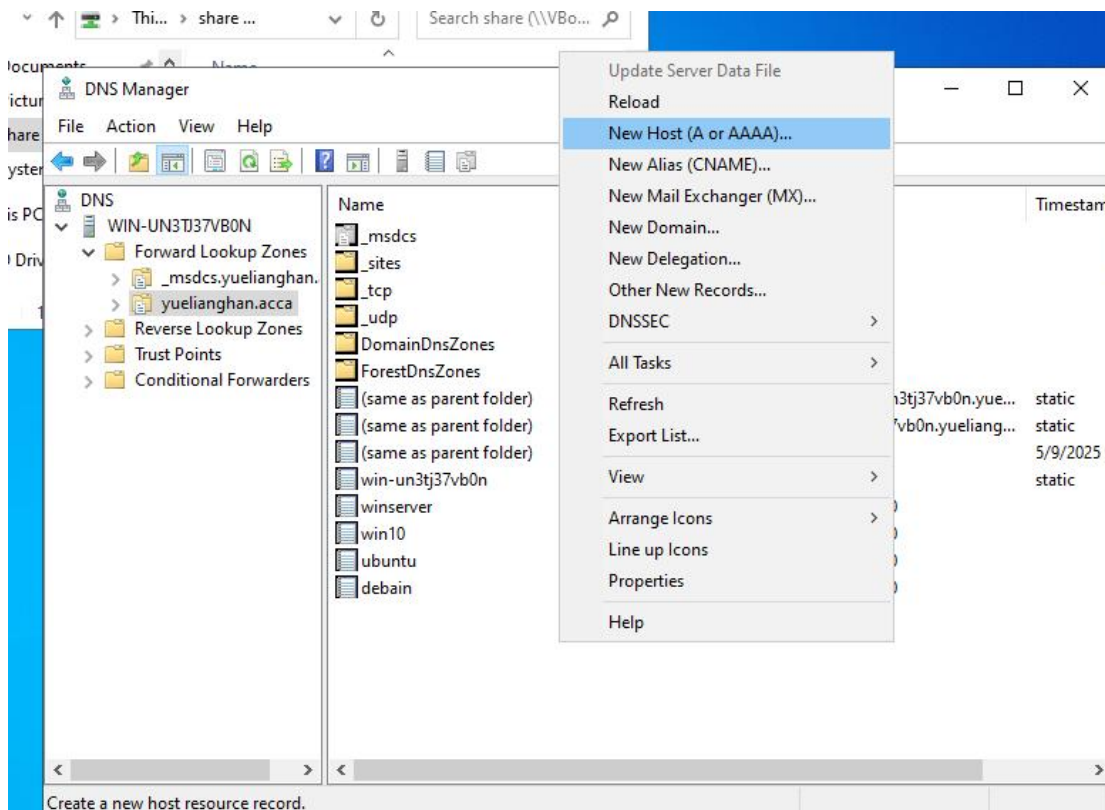
```
Administrator: Windows PowerShell

PS C:\Users\Administrator> Get-ADForest

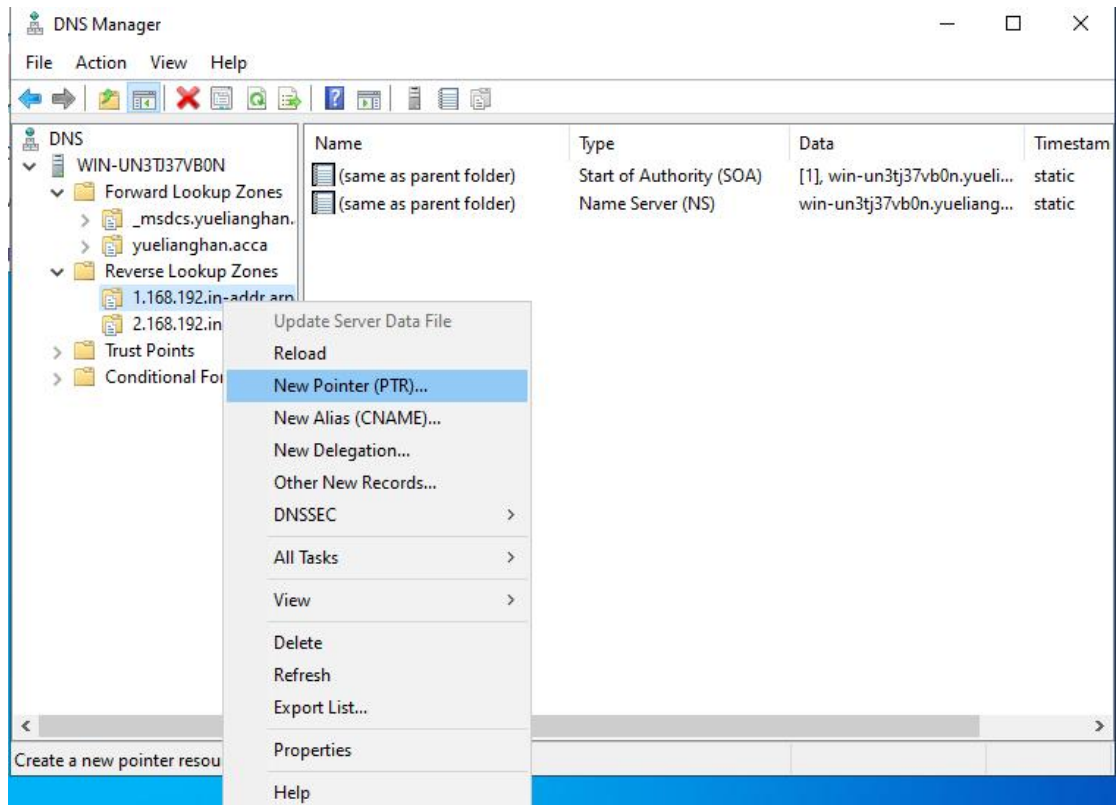
ApplicationPartitions : {DC=ForestDnsZones,DC=yuelianghan,DC=acca, DC=DomainDnsZones,DC=yuelianghan,DC=acca}
CrossForestReferences : {}
DomainNamingMaster    : WIN-UN3TJ37VB0N.yuelianghan.acca
Domains               : {yuelianghan.acca}
ForestMode            : Windows2016Forest
GlobalCatalogs       : {WIN-UN3TJ37VB0N.yuelianghan.acca}
Name                  : yuelianghan.acca
PartitionsContainer   : CN=Partitions,CN=Configuration,DC=yuelianghan,DC=acca
RootDomain            : yuelianghan.acca
SchemaMaster          : WIN-UN3TJ37VB0N.yuelianghan.acca
Sites                 : {Default-First-Site-Name}
SPNSuffixes           : {}
UPNSuffixes           : {}

PS C:\Users\Administrator> D_
```

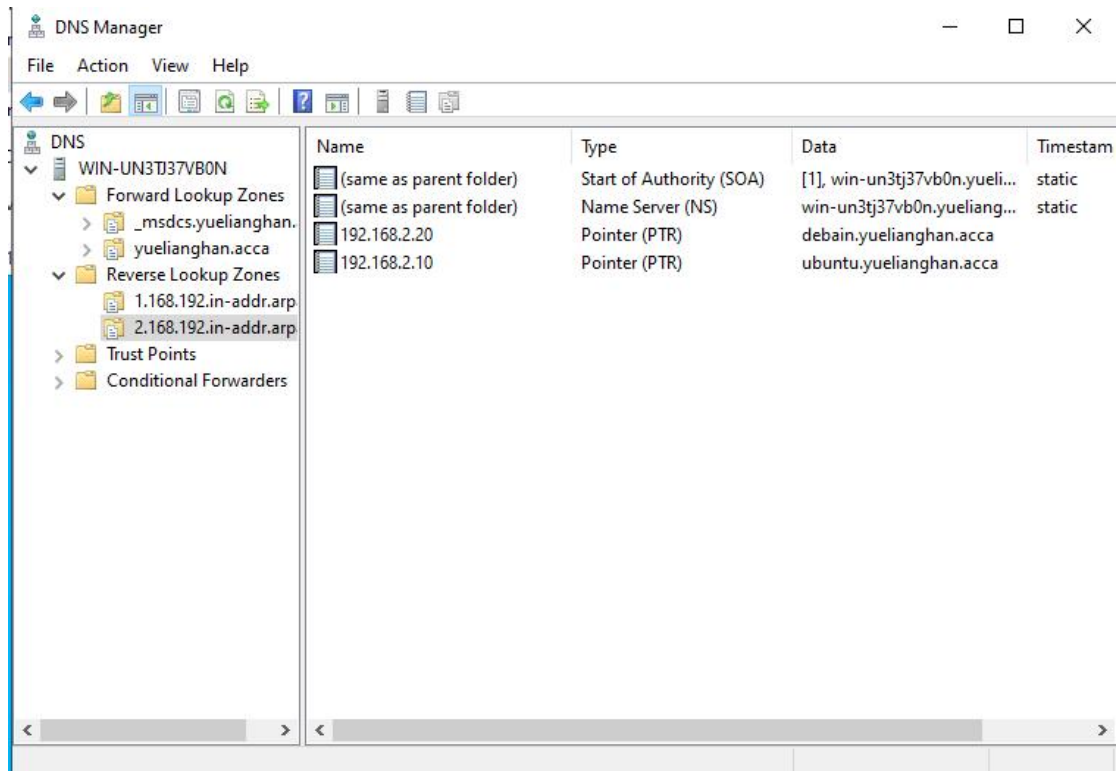
1.3.2



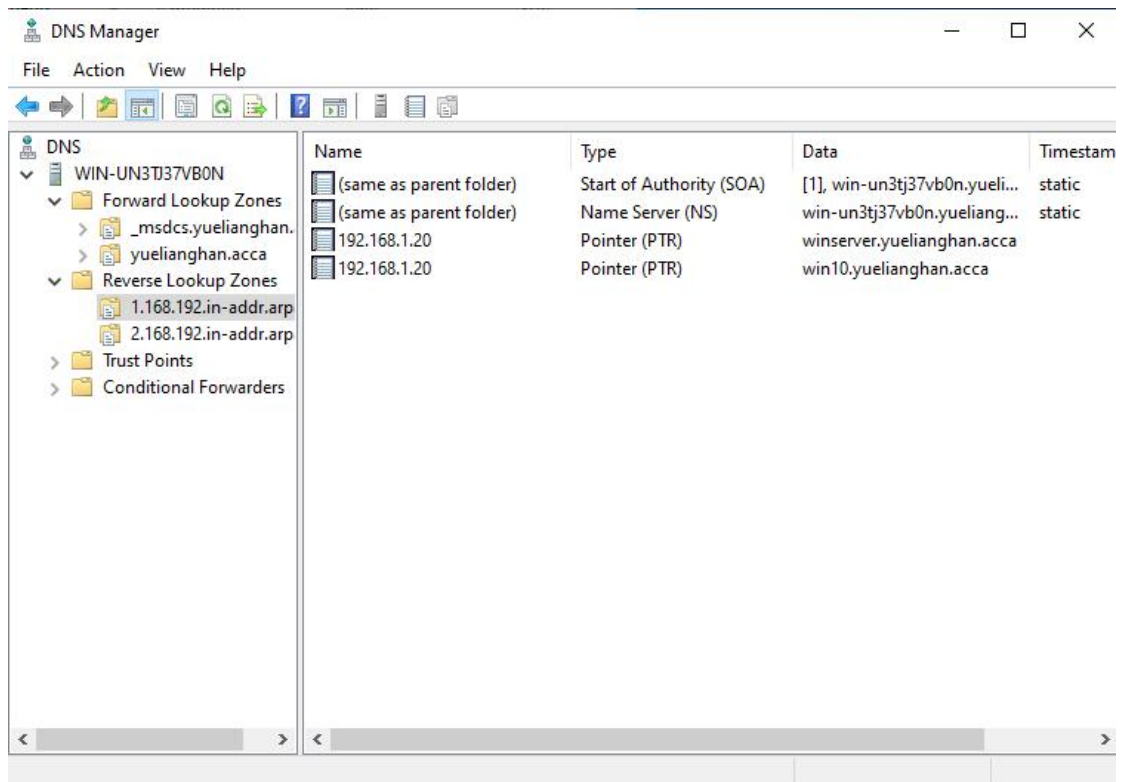
1.4



1.5.1



1.5.2



1.5.3

```

PS C:\Users\Administrator> Import-Module ActiveDirectory
>> Import-Csv "Z:\users_list_updated.csv" | Where-Object {$_.domain -eq "YOURNAME.ACCA"} | ForEach-Object {
>>     $first = $_.first_name
>>     $last = $_.last_name
>>     $sam = "$($first.Substring(0,1)).$last".ToLower()
>>     $name = "$first $last"
>>     $supn = "$sam@yuelianghan.acca"
>>
>>     New-ADUser `
>>         -Name $name `
>>         -GivenName $first `
>>         -Surname $last `
>>         -SamAccountName $sam `
>>         -UserPrincipalName $supn `
>>         -EmailAddress $supn `
>>         -Title $_.job `
>>         -Department $_.department `
>>         -AccountPassword (ConvertTo-SecureString "P@ssw0rd123" -AsPlainText -Force) `
>>         -Enabled $true `
>>         -Path "CN=Users,DC=yuelianghan,DC=acca" `
>>         -ChangePasswordAtLogon $true
>> }
New-ADUser : The operation failed because UPN value provided for addition/modification is not unique forest-wide
At line:10 char:5
+ New-ADUser `
+ ~~~~~
+ CategoryInfo          : NotSpecified (CN=Angela Johns...ianghan,DC=acca:String) [New-ADUser], ADException
+ FullyQualifiedErrorId : ActiveDirectoryServer:8648,Microsoft.ActiveDirectory.Management.Commands.NewADUser
  
```

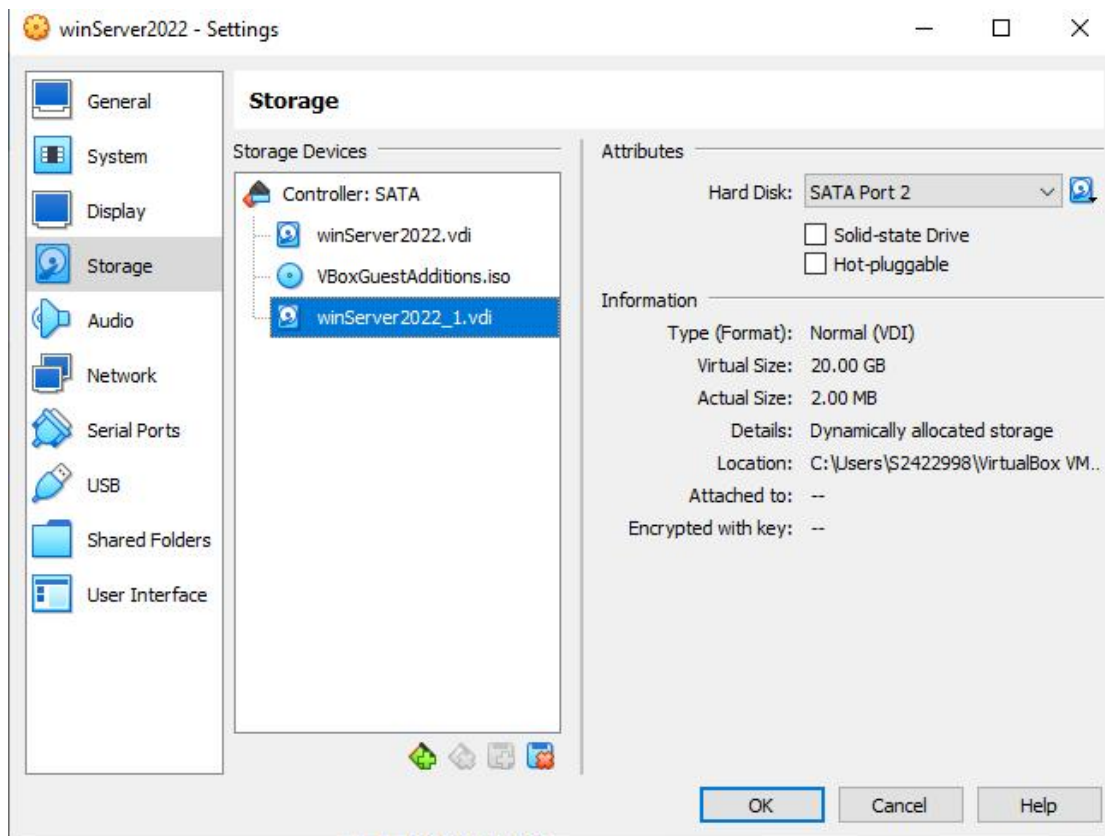
1.6.1

```

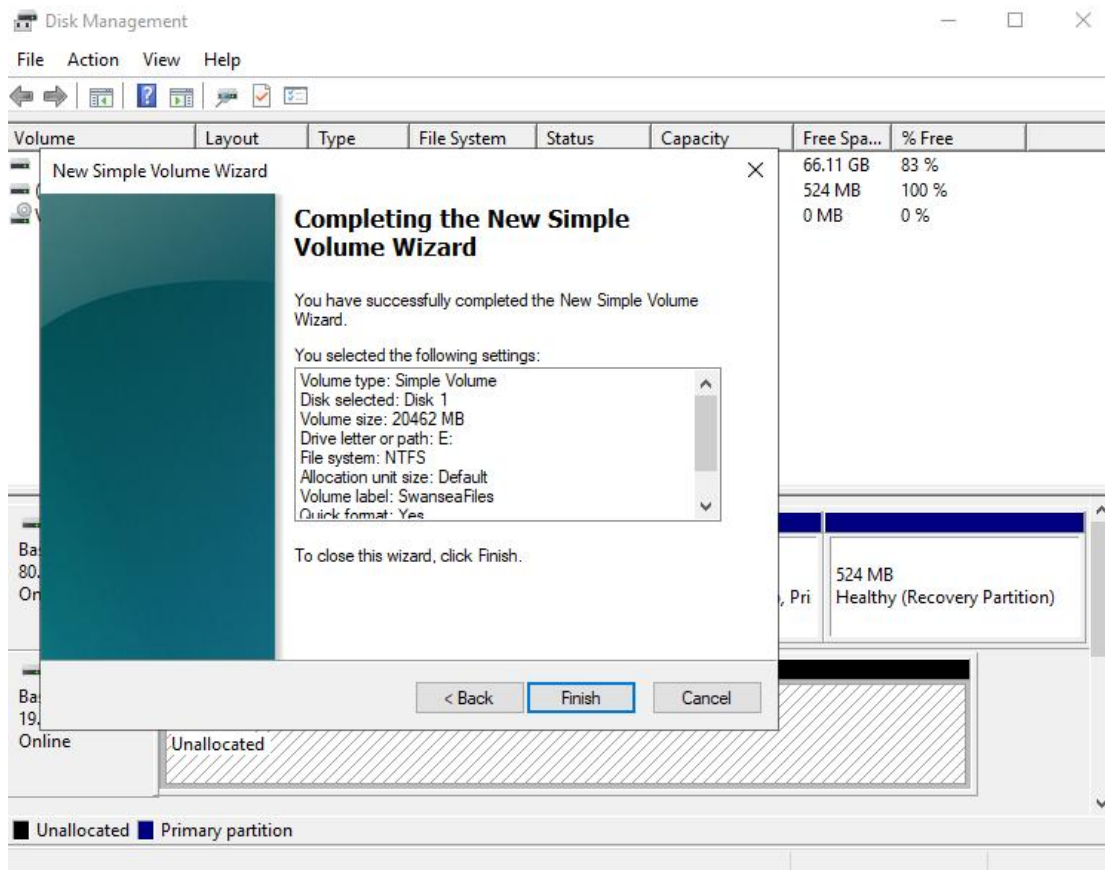
PS C:\Users\Administrator> Get-ADUser -Filter * -SearchBase "CN=Users,DC=yuelianghan,DC=acca" | Select Name,SamAccountName
Name                               SamAccountName
-----
Administrator                      Administrator
Guest                              Guest
krbtgt                             krbtgt
Brian White                        b.white
Susan Jackson                      s.jackson
Sherri Carney                      s.carney
Jonathon Doyle                    j.doyle
Carla Harrington                  c.harrington
Billy Williams                    b.williams
Donna Clarke                      d.clarke
Kelsey Garcia                     k.garcia
Wesley Cole                       w.cole
Steven Mcgee                      s.mcgee
Sara Tran                         s.tran
Carrie Walker                     c.walker
Cody Salinas                      c.salinas
Heather Middleton                 h.middleton
Thomas Gutierrez                  t.gutierrez
Patrick Hoffman                   p.hoffman
James Anderson                    j.anderson
Colleen Wall                      c.wall
Rachel Martin                     r.martin
Alexis Mcdaniel                   a.mcdaniel
Eric Hall                         e.hall

```

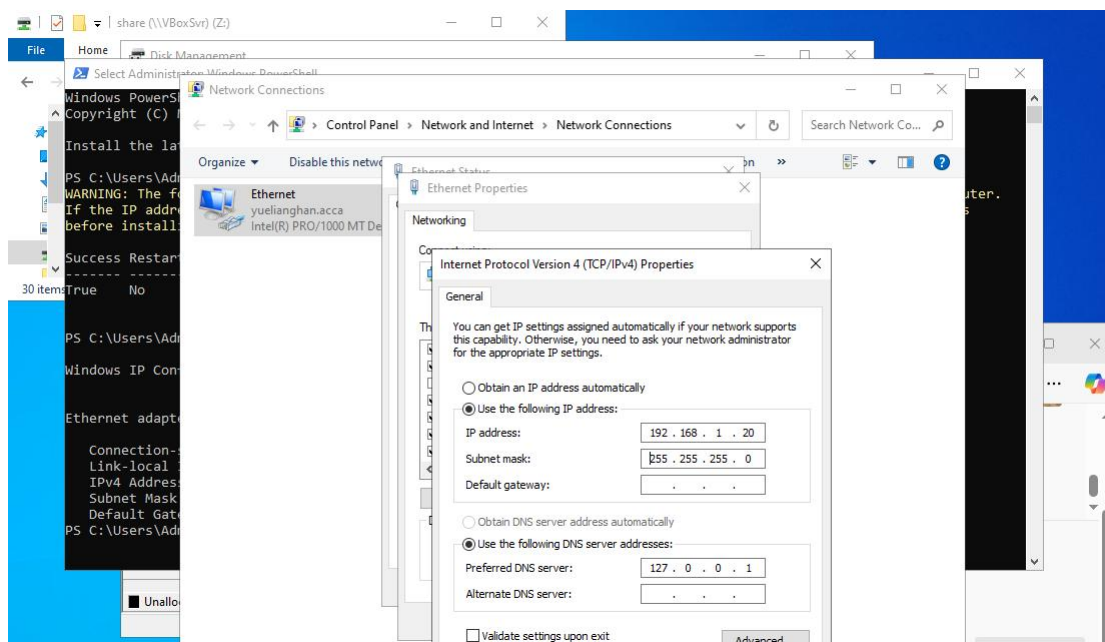
1.6.2



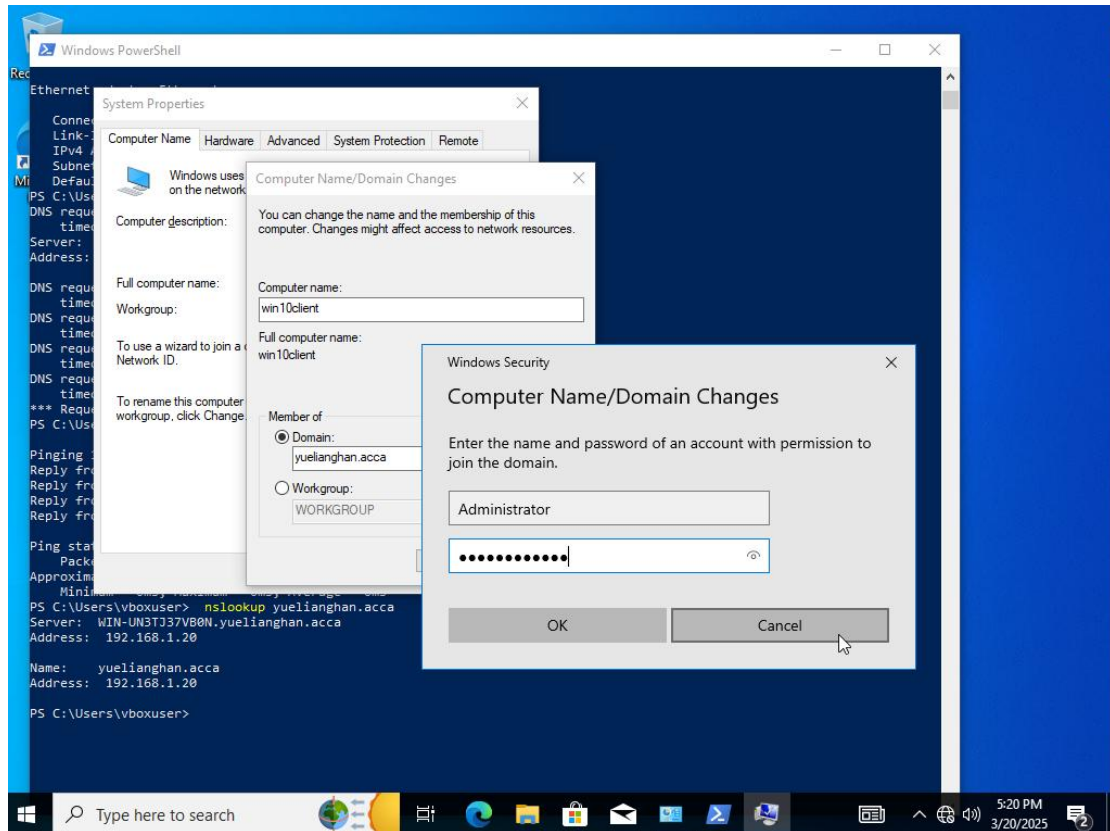
1.7.1



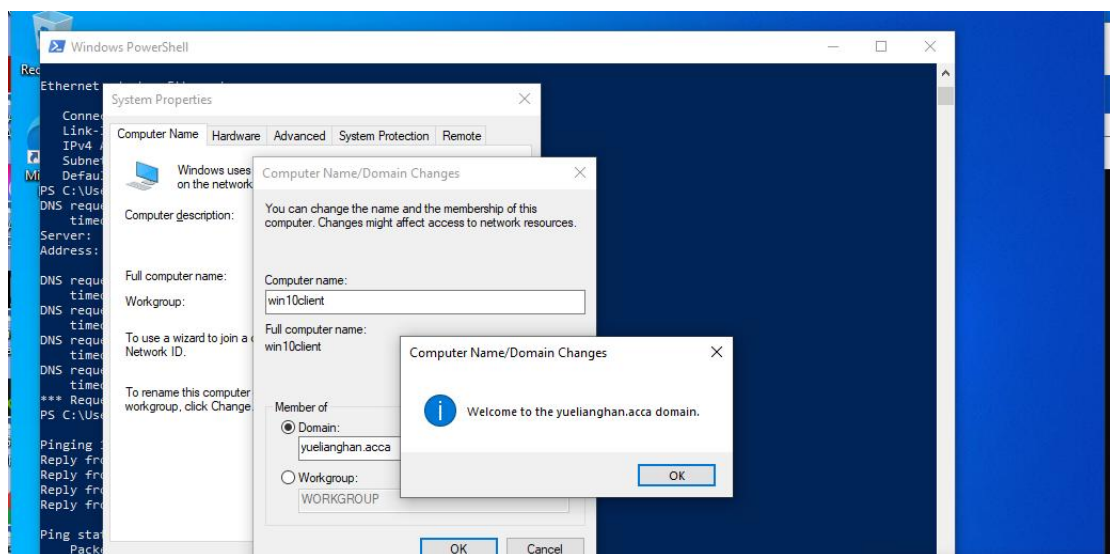
1.7.2



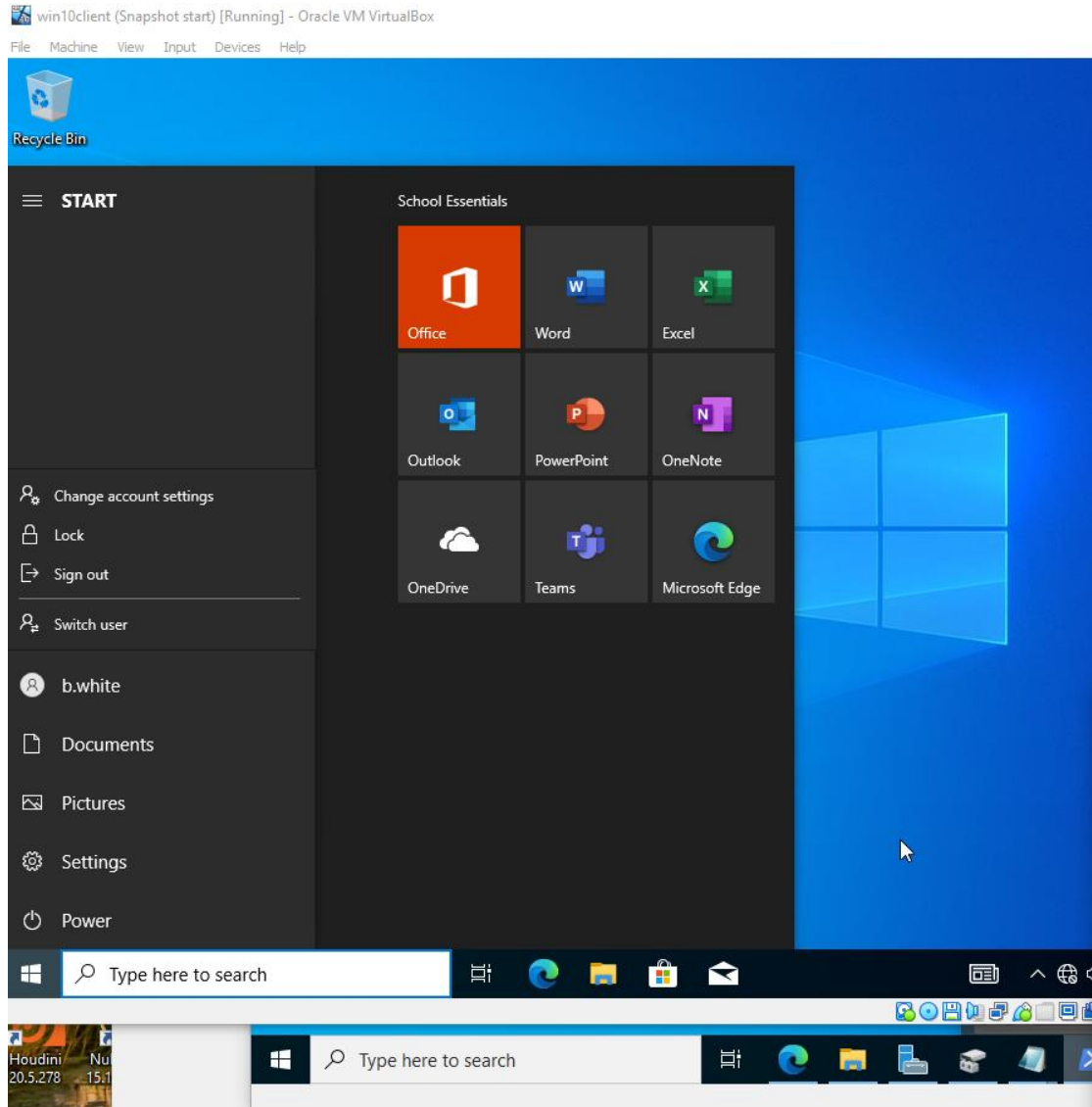
1.8



1.9.1



1.9.2



```
vboxuser@debain: ~  
vboxuser@debain:~$ su -  
Password:  
root@debain:~# udo apt update  
sudo apt install isc-dhcp-server  
-bash: udo: command not found  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following additional packages will be installed:  
  policycoreutils selinux-utils  
Suggested packages:  
  policykit-1 isc-dhcp-server-ldap ieee-data  
The following NEW packages will be installed:  
  isc-dhcp-server policycoreutils selinux-utils  
0 upgraded, 3 newly installed, 0 to remove and 46 not upgraded.  
Need to get 1,766 kB of archives.  
After this operation, 7,818 kB of additional disk space will be used.  
Do you want to continue? [Y/n] Y  
Ign:1 http://deb.debian.org/debian bookworm/main amd64 isc-dhcp-server amd64 4.4.3-P1-2  
Ign:2 http://deb.debian.org/debian bookworm/main amd64 selinux-utils amd64 3.4-1+b6  
Ign:3 http://deb.debian.org/debian bookworm/main amd64 policycoreutils amd64 3.4-1  
Ign:1 http://deb.debian.org/debian bookworm/main amd64 isc-dhcp-server amd64 4.4.3-P1-2  
Ign:2 http://deb.debian.org/debian bookworm/main amd64 selinux-utils amd64 3.4-1+b6  
Ign:3 http://deb.debian.org/debian bookworm/main amd64 policycoreutils amd64 3.4-1  
Ign:1 http://deb.debian.org/debian bookworm/main amd64 isc-dhcp-server amd64 4.4.3-P1-2  
Ign:2 http://deb.debian.org/debian bookworm/main amd64 selinux-utils amd64 3.4-1+b6  
Ign:3 http://deb.debian.org/debian bookworm/main amd64 policycoreutils amd64 3.4-1  
Err:1 http://deb.debian.org/debian bookworm/main amd64 isc-dhcp-server amd64 4.4.3-P1-2
```

2.1.1

```
vboxuser@debain: ~  
vboxuser@debain:~$ su -  
Password:  
root@debain:~# apt update  
apt install strongswan -y  
gn:1 http://security.debian.org/debian-security bookworm-security InRelease  
gn:2 http://deb.debian.org/debian bookworm InRelease  
gn:3 http://deb.debian.org/debian bookworm-updates InRelease  
gn:1 http://security.debian.org/debian-security bookworm-security InRelease  
gn:2 http://deb.debian.org/debian bookworm InRelease  
gn:3 http://deb.debian.org/debian bookworm-updates InRelease  
gn:1 http://security.debian.org/debian-security bookworm-security InRelease  
gn:2 http://deb.debian.org/debian bookworm InRelease  
gn:3 http://deb.debian.org/debian bookworm-updates InRelease  
rr:1 http://security.debian.org/debian-security bookworm-security InRelease  
Temporary failure resolving 'security.debian.org'  
rr:2 http://deb.debian.org/debian bookworm InRelease  
Temporary failure resolving 'deb.debian.org'  
rr:3 http://deb.debian.org/debian bookworm-updates InRelease  
Temporary failure resolving 'deb.debian.org'  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
6 packages can be upgraded. Run 'apt list --upgradable' to see them.  
E: Failed to fetch http://deb.debian.org/debian/dists/bookworm/InRelease Temporary
```

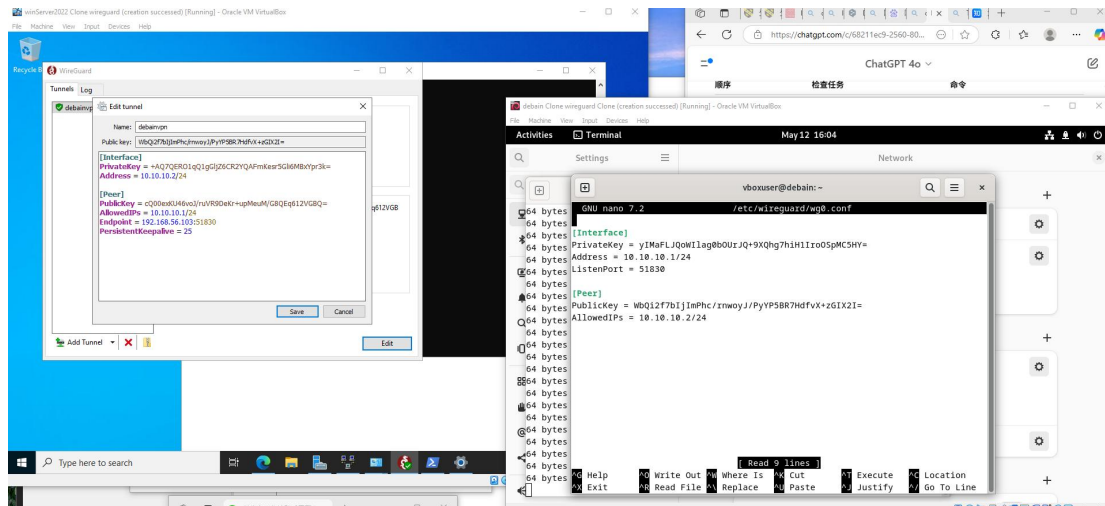
2.1.2

```
vboxuser@debain: ~
Hit:1 http://deb.debian.org/debian bookworm InRelease
Hit:2 http://security.debian.org/debian-security bookworm-security InRelease
Hit:3 http://deb.debian.org/debian bookworm-updates InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
46 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
iptables is already the newest version (1.8.9-2).
0 upgraded, 0 newly installed, 0 to remove and 46 not upgraded.
root@debain:~# iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
iptables -A OUTPUT -p icmp --icmp-type echo-reply -j ACCEPT
root@debain:~#
root@debain:~#
root@debain:~#
root@debain:~# iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
iptables -A OUTPUT -p icmp --icmp-type echo-reply -j ACCEPT
root@debain:~# iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
iptables -A OUTPUT -p icmp --icmp-type echo-reply -j ACCEPT
root@debain:~# iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
root@debain:~#
```

2.1.3

```
vboxuser@debain: ~
root@debain:~# sudo apt update
sudo apt install ptp-linux network-manager-pttp network-manager-pttp-gnome -y
Hit:1 http://security.debian.org/debian-security bookworm-security InRelease
Hit:2 http://deb.debian.org/debian bookworm InRelease
Hit:3 http://deb.debian.org/debian bookworm-updates InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
46 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  network-manager-pttp network-manager-pttp-gnome ptp-linux
0 upgraded, 3 newly installed, 0 to remove and 46 not upgraded.
Need to get 187 kB of archives.
After this operation, 996 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian bookworm/main amd64 ptp-linux amd64 1.10.0-1
[55.9 kB]
Get:2 http://deb.debian.org/debian bookworm/main amd64 network-manager-pttp amd6
4 1.2.12-1 [98.4 kB]
Get:3 http://deb.debian.org/debian bookworm/main amd64 network-manager-pttp-gnom
e amd64 1.2.12-1 [32.4 kB]
Fetched 187 kB in 0s (2,529 kB/s)
```

2.2.1

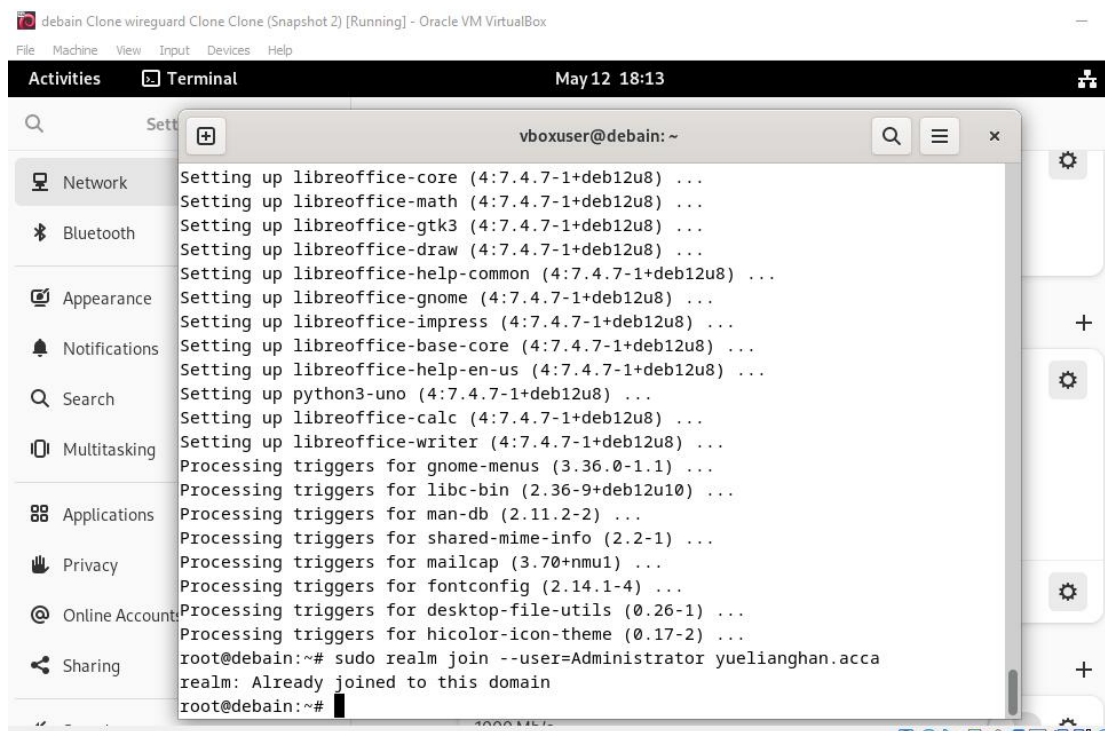


2.2.2

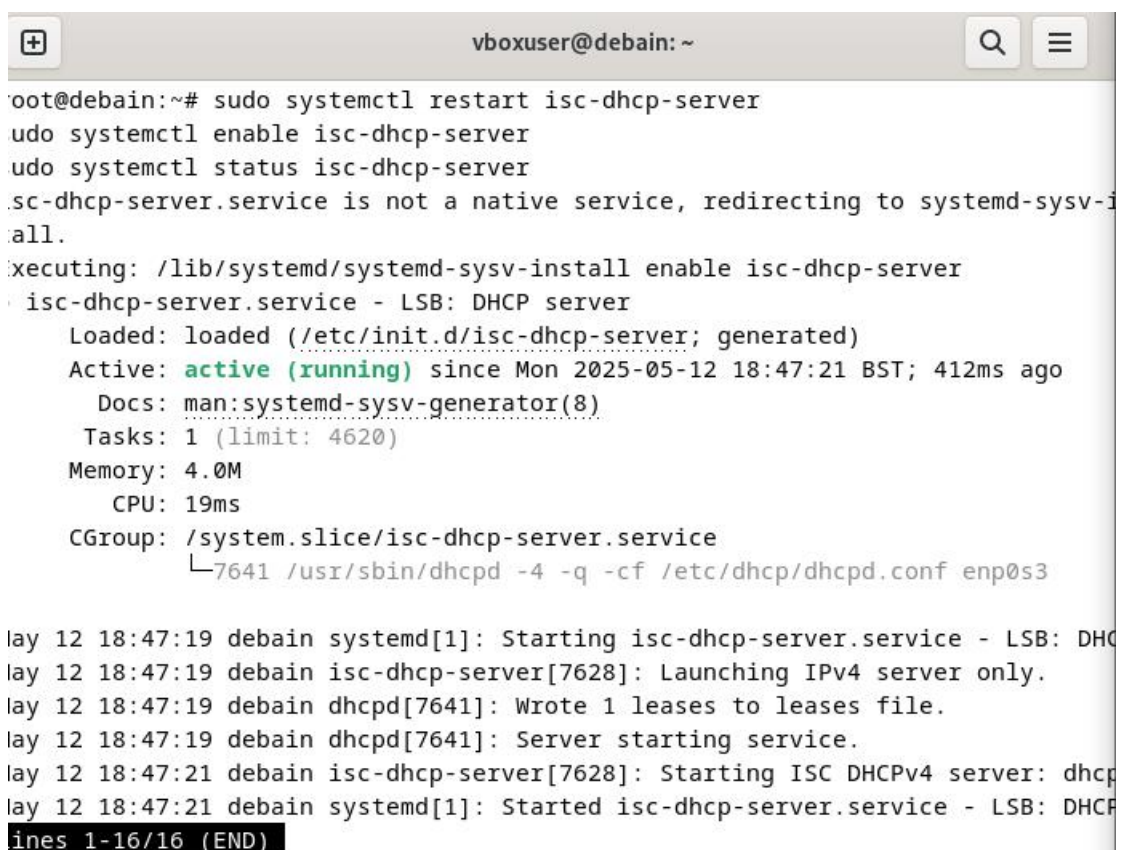
```
vboxuser@debain: ~
Name: yuelianghan.acca
Address: 192.168.56.104

root@debain:~# realm discover yuelianghan.acca
yuelianghan.acca
  type: kerberos
  realm-name: YUELIANGHAN.ACCA
  domain-name: yuelianghan.acca
  configured: no
  server-software: active-directory
  client-software: sssd
  required-package: sssd-tools
  required-package: sssd
  required-package: libnss-sss
  required-package: libpam-sss
  required-package: adcli
  required-package: samba-common-bin
root@debain:~# sudo realm join --user=Administrator yuelianghan.acca
Password for Administrator:
* Installing necessary packages: sssd-tools
See: journalctl REALMD_OPERATION=r21250.12168
realm: Couldn't join realm: Cannot download packages whilst offline
root@debain:~# realm list
root@debain:~#
```

2.3.1



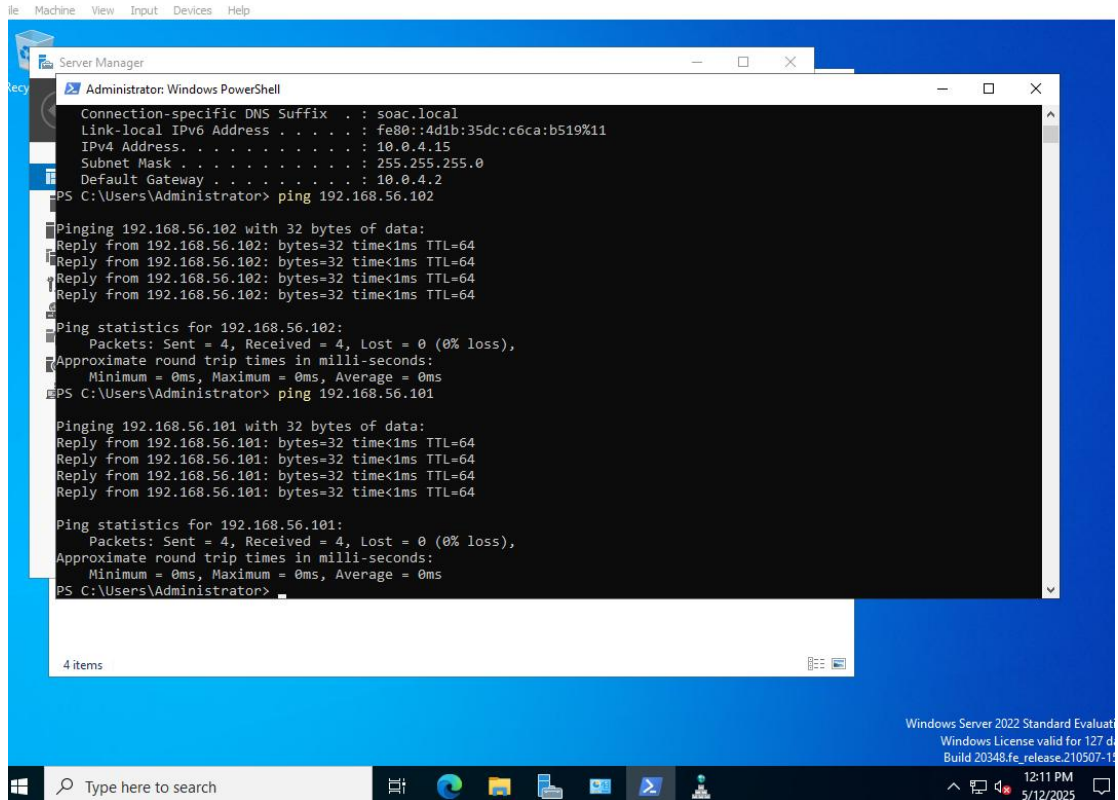
2.3.2



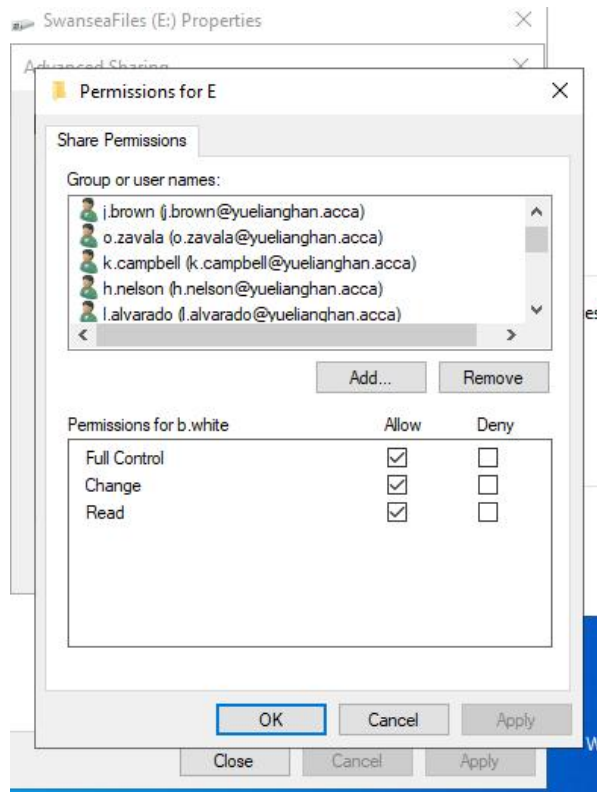
2.4

```
debain@debain-VirtualBox: ~  
debain@debain-VirtualBox:~$ realm list  
yuelianghan.acca  
type: kerberos  
realm-name: YUELIANGHAN.ACCA  
domain-name: yuelianghan.acca  
configured: kerberos-member  
server-software: active-directory  
client-software: sssd  
required-package: sssd-tools  
required-package: sssd  
required-package: libnss-sss  
required-package: libpam-sss  
required-package: adcli  
required-package: samba-common-bin  
login-formats: %U@yuelianghan.acca  
login-policy: allow-realm-logins  
debain@debain-VirtualBox:~$ sudo realm join --user=Administrator yuelianghan.acca  
sudo] password for debain:  
realm: Already joined to this domain  
debain@debain-VirtualBox:~$
```

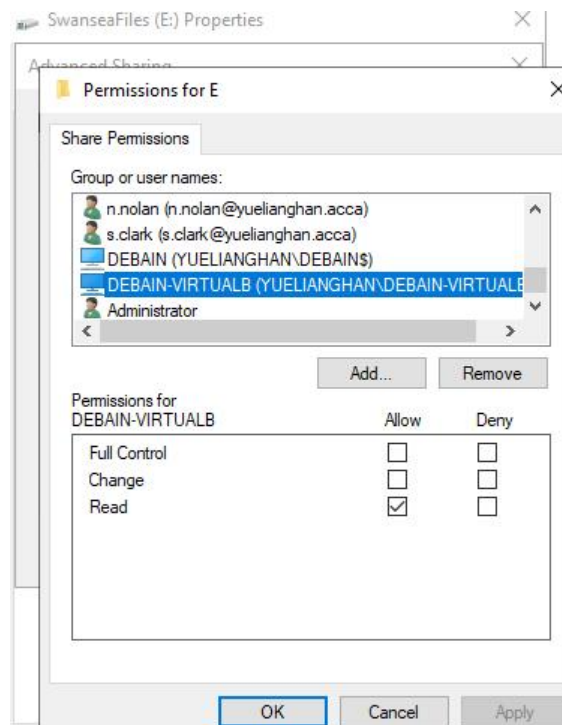
2.5



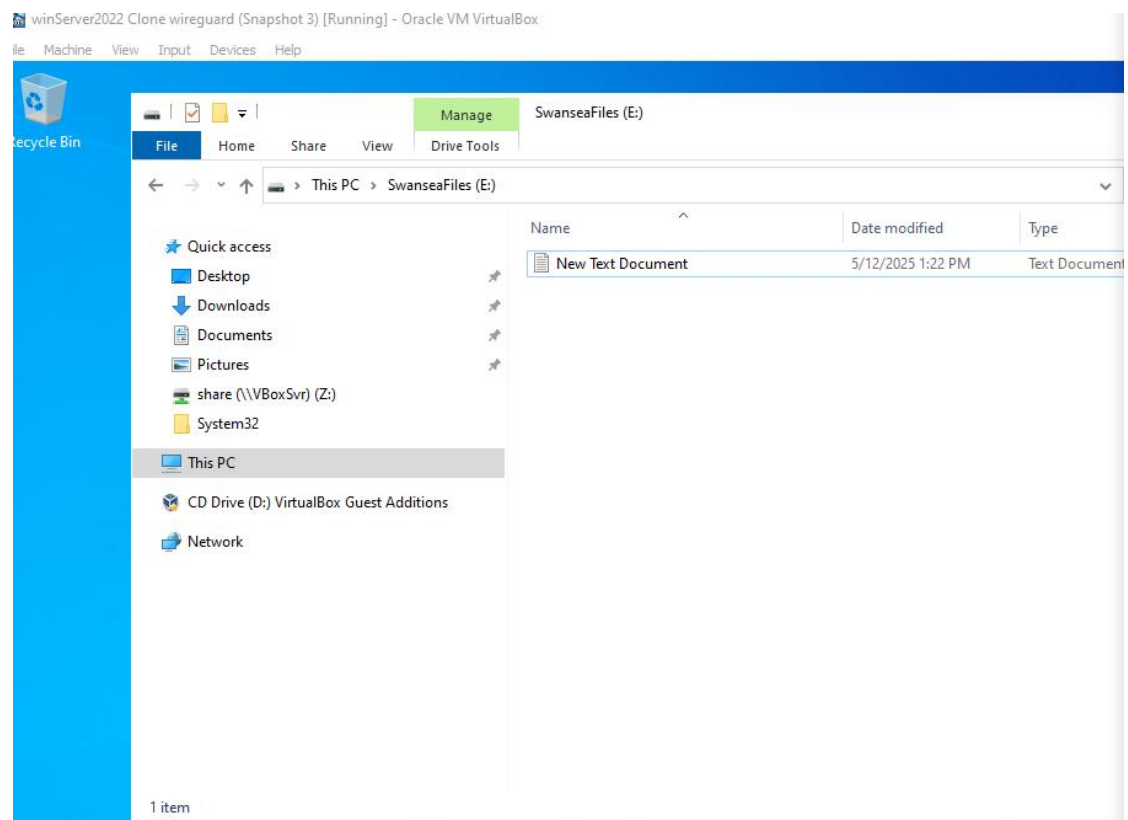
3



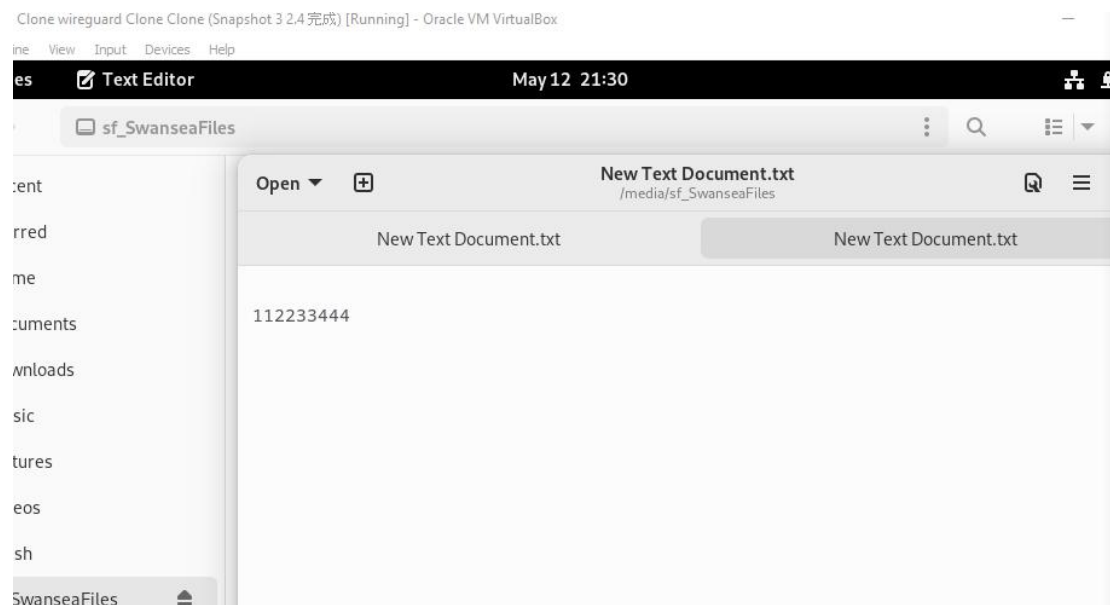
4.1



4.2



4.3



4.4