

NOSS | Coursework n°1

Daligaud Ewenn

Table of contents

Table of figures	4
Abstract	5
Introduction	6
TASK 1 Research Report	7
Research on hypervisor	8
Introduction	8
Architecture of hypervisors.....	9
Hardware requirements.....	11
Pros and Cons	12
IP addressing	13
Introduction	13
How data travel	14
How routers work	15
How switches work.....	16
IP addresses in networks	17
Research on routing	18
Introduction	18
Internal Gateway Protocol (IGP)	19
External Gateway Protocols (EGP).....	20
Research on network services	21
Virtual Private network (VPN)	21
Active Directory Domain Services (ADDS).....	22
Dynamic Host Configuration Protocol (DHCP).....	23
TASK 2 Technical Documentation	24
2.1 Choosing the Hypervisor	25
2.2 Download and Setup	26
2.3 Configure IP Addressing.....	31
2.4 Testing and Troubleshooting	33
TASK 3 Reflection	40
3.1 Technical Reflection	41
3.2 Personal Reflection	41
Conclusion.....	42
Appendix	43
Classes of IP addresses	43

NAT	43
Autonomous Systems	43
Sudo.....	43
References	44

Table of figures

Figure 1 : Global representation of how hypervisors work	8
Figure 2 : Representation of the architecture of type 1 hypervisors.....	9
Figure 3 : Representation of the architecture of type 2 hypervisors.....	10
Figure 4 : Example of an IP address by typing "ipconfig" in the command prompt ...	13
Figure 5 : Composition of packets	14
Figure 6 : public and private IP addresses.....	15
Figure 7 : Switches making the link between devices and the router	16
Figure 8 : Example of IPv4 & IPv6 addresses by typing "ipconfig" in the command prompt	17
Figure 9 : Organization of OSPF networks	19
Figure 10 : Path to "www.google.com" using BGP by typing "tracert + website's URL"	20
Figure 11 : How VPNs work.....	21
Figure 12 : Representation of Active Domain Directory Services (ADDS)	22
Figure 13 : How DHCP works.....	23
Figure 14 : Oracle Virtual Box main menu	26
Figure 15 : Downloading the Windows Server 2022 ISO.....	27
Figure 16 : Creating the Windows Server 2022 Virtual Machine	27
Figure 17 : Selecting the Operating System	28
Figure 18 : Configuring storage locations	28
Figure 19 : Download Linux from official Debian website	29
Figure 20 : Debian Linux setup.....	29
Figure 21 : Downloading Ubuntu from official Ubuntu website	30
Figure 22 : Configuring the Client IP Addressing (Windows 10)	31
Figure 23 : Configuring the server IP Addressing (Windows Server 2022)	31
Figure 24 : Configuring the IP addressing in the "interfaces" file, Linux Server (Debian Linux).....	32
Figure 25 : Configuring the IP addressing of the Linux Client (Ubuntu)	32
Figure 26 : Pinging the Server (S-DC) from the Client (ClientS1) (not working)	33
Figure 27 : Pinging the Client (ClientS1) from the Server (S-DC) (not working)	33
Figure 28 : Disabling all the Client's firewall profiles	34
Figure 29 : Pinging the Client (ClientS1) from the server (S-DC) (working).....	35
Figure 30 : Allowing ICMP packets on the Server.....	36
Figure 31 : Pinging the Server (S-DC) from the Client (ClientS1) (working)	36
Figure 32 : Pinging Debian (C-Server) from Windows (S-DC).....	37
Figure 33 : Pinging Windows (S-DC) from Linux (C-Server).....	37
Figure 34 : Pinging Ubuntu (ClientC1) from Debian (C-Server).....	38
Figure 35 : Pinging the Debian Linux (C-Server) from Ubuntu (ClientC1)	38

Abstract

This coursework is divided into three tasks, the first one is a report of the research we did, the second is about the installation and configuration of our virtual machines, then the third task is a personal reflection covering the work we have done.

In the first task, we will be looking at some of the key points to understand the concept of virtualization. We will start by doing detailed research about hypervisors and how it works so we can properly use it to create our virtual machines and then our network in the second task. To create a network, we need to know about networking and all the protocols associated to it, that is why we are going to have a look at some of them like Internal/External Gateway Protocols (IGP/EGP), Dynamic Host Configuration Protocol (DHCP), Internet Protocol (IP), but also some services like Virtual Private Network (VPN) and Active Directory Domain Services (ADDS).

In the second task we will be creating our Network on the different virtual machines we have to install. We will start by choosing the hypervisor we want to use and install the different virtual machines on it. Then we will configure the network as per the given topology and some of the protocols and services seen in the first task.

Then, the third task consists of writing a review of the projects, that includes presenting the difficulties/challenges we faced and documenting it and writing a personal reflection about the project in its globality.

Introduction

In today's digital world, virtualization is a significant technology for optimizing computing resources, security and networking. As organizations and companies increasingly rely on virtualized systems to handle their heavy workload, it is a skill that must be acquired to deploy and manage virtual machines effectively. This project aims to explore the theoretical and practical aspects of virtualization with the final goal being the creation of a virtual network with several services.

Moreover, this project is also an opportunity to develop a very good understanding of how virtualization and networking benefits IT infrastructures in these modern times. With the study of networking protocols, virtual environment configuration, and the consideration of the multiple problems encountered, this is a great and interesting way to acquire knowledge that can be directly used in the real-world IT administration scenarios.

TASK 1 | Research Report

Research on hypervisor

Introduction

Virtualization is the process that uses software to “*enable multiple operating systems instances to execute simultaneously on a single physical system*” [1]. The software is called a hypervisor, it is a layer that runs above your server that manages the resources division and allows you to split up the power of your server (host) and spread it out among as many virtual machines your server can provide power to.

Two types of hypervisors are available in the market:

- Type 1 hypervisors are used in datacenters and in heavy workload situations in companies.
- Type 2 hypervisors are mostly used for testing purposes as they are portable because it only requires one computer. It is used by users who need to simulate many operating systems.

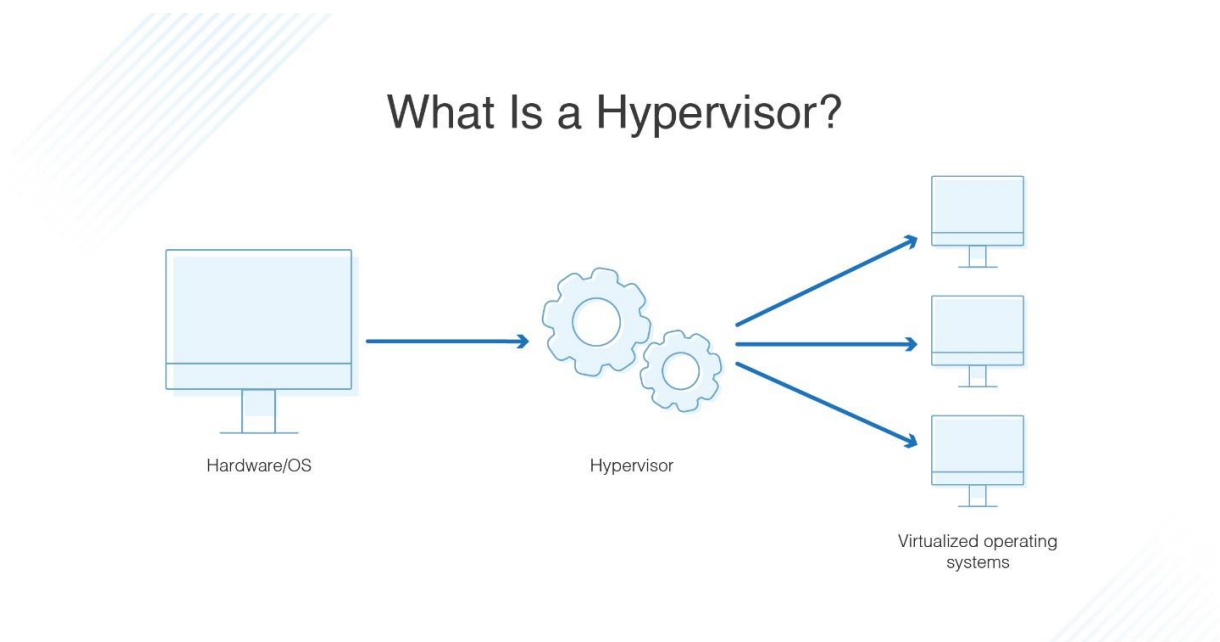


Figure 1 : Global representation of how hypervisors work

Architecture of hypervisors

Architecture of type 1 hypervisors

The type 1 hypervisor has direct access to the hardware resources, and it manages the allocation of resources to each virtual machine in the network. It is directly installed on the physical server and does not require the use of the operating system.

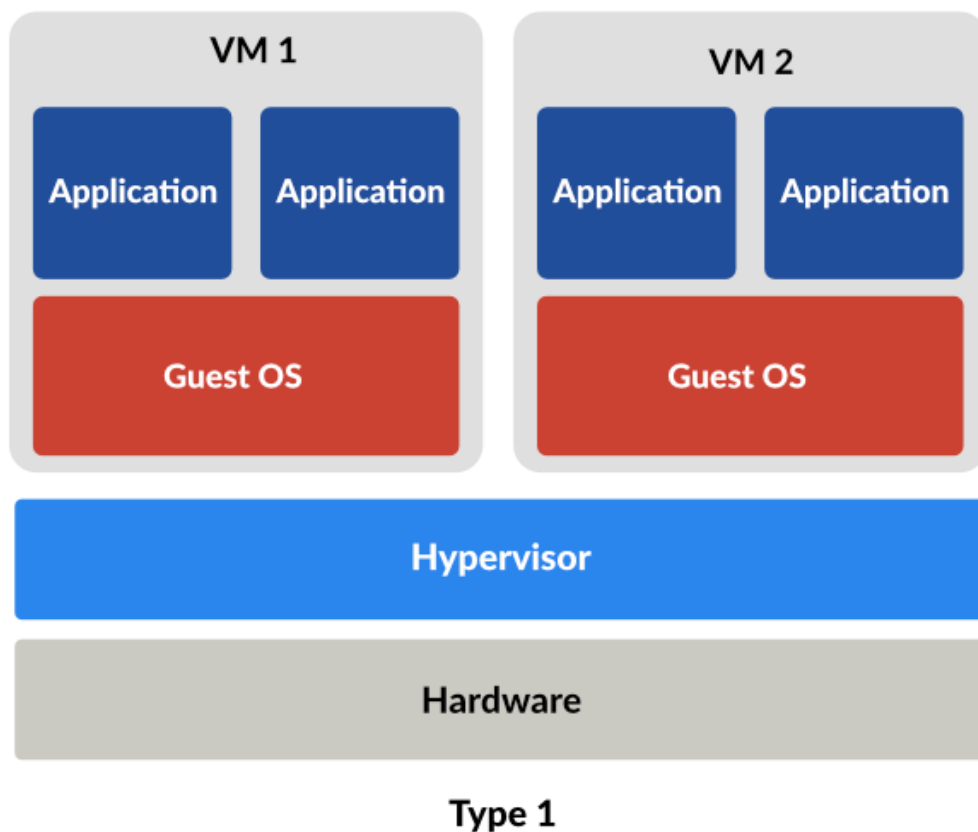


Figure 2 : Representation of the architecture of type 1 hypervisors

Architecture of type 2 hypervisors

The type 2 hypervisor is installed on the OS of the host, it does not have direct access to hardware resources, it has to negotiate the allocation of resources with the host OS but the host OS gives priority to its own functions and applications.

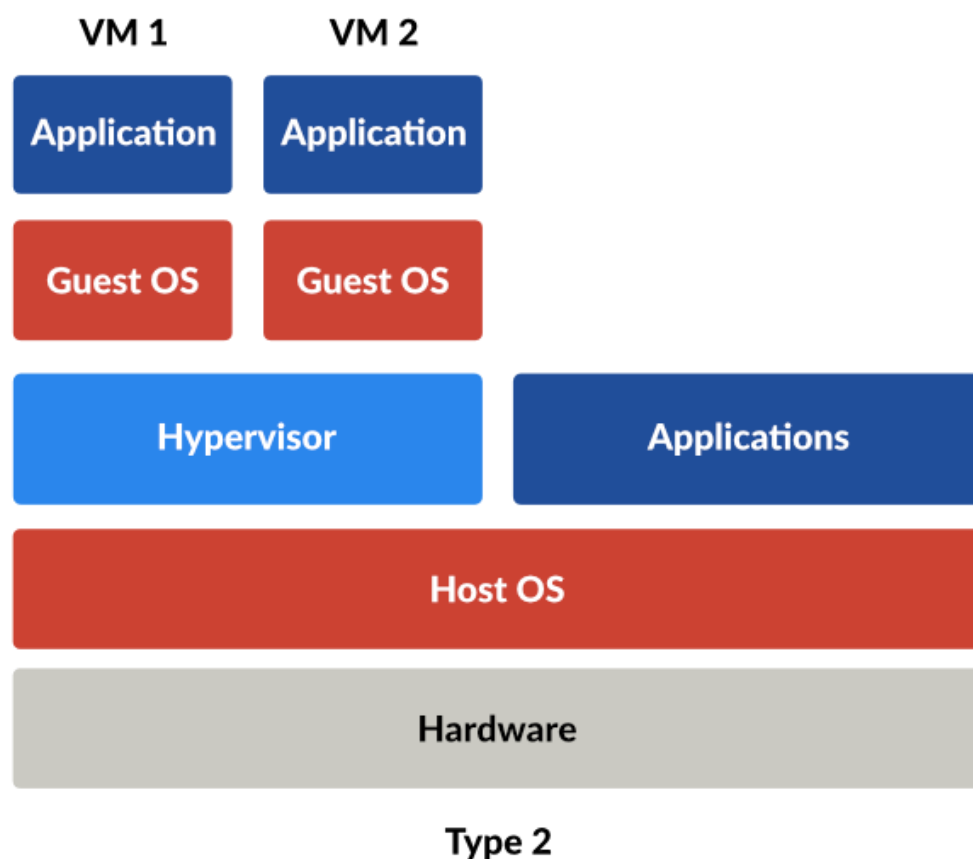


Figure 3 : Representation of the architecture of type 2 hypervisors

Hardware requirements

To run a virtual machine, a computer must have some minimum hardware capacity. For type 1 hypervisors, we will take as reference Microsoft Hyper-V, according to Microsoft website. For type 2 hypervisors, we will take as reference Oracle Virtual Box according to Oracle Website.

	Type 1	Type 2
Processor	Any 64-bit processor	Any recent Intel or AMD processor
RAM	At least 4GB	
Storage	At least 5BG of storage	
Operating System	Windows 10 Pro/Enterprise, Windows 11 Pro/Enterprise	Windows 8.1 or newer

Pros and Cons

Type 1 hypervisors are the most secure, the lowest on latency and the most used, but they are more expensive. Some examples can be VMware ESXI, Microsoft Hyper-V.

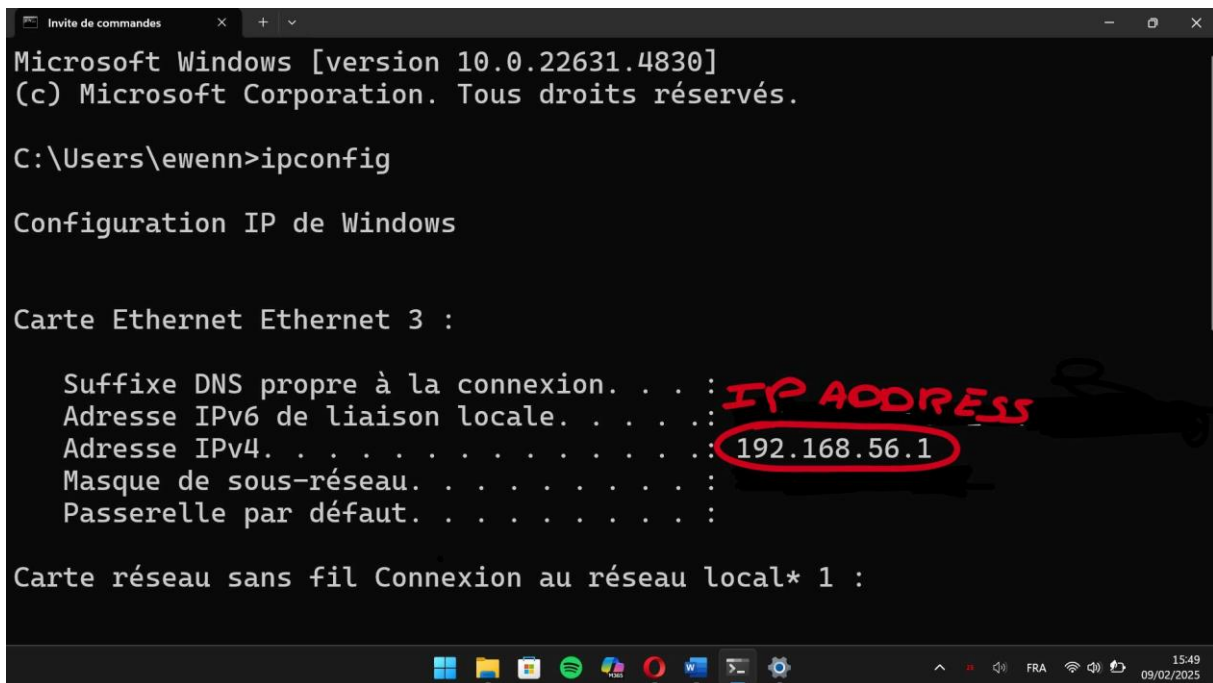
By allowing multiple virtual machines to run independently on a single physical server, hypervisors help maximize resources utilization and reduce infrastructure costs.

Hypervisors provide strong isolation between virtual machines, ensuring that each VM can operate independently with its own dedicated resources without affecting others.

IP addressing

Introduction

In a network, computers are linked together, and all networks are linked together, that creates the internet. Computers from a network can communicate with computers from any other network but they must know who they are communicating with, and this is where IP addresses come in. An IP address, or Internet Protocol Address is a series of numbers that identifies any connected device.



```
Microsoft Windows [version 10.0.22631.4830]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\ewenn>ipconfig

Configuration IP de Windows

Carte Ethernet Ethernet 3 :

    Suffixe DNS propre à la connexion. . . . : 
    Adresse IPv6 de liaison locale. . . . . : 
    Adresse IPv4. . . . . : 192.168.56.1
    Masque de sous-réseau. . . . . : 
    Passerelle par défaut. . . . . : 

Carte réseau sans fil Connexion au réseau local* 1 :
```

Figure 4 : Example of an IP address by typing “ipconfig” in the command prompt

Each number separated by dots is called an “octet”.

There are 5 classes of IP addresses, [here](#) is a table that explains each of them.

How data travel

Basically, our computer sends data to internet servers which then sends the data to the computer we want to send the data to. When data is transferred, it is broken up into small packages called “packets” that all contains the same number of bits. Each packet contains the “header”. *“The headers usually include addressing and routing information”* [2] such as the IP address.

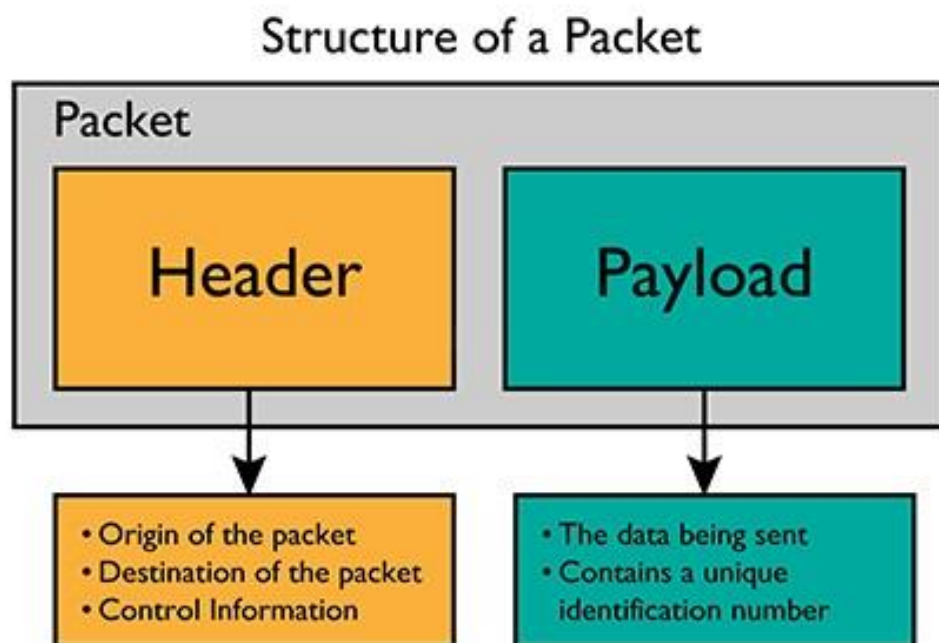


Figure 5 : Composition of packets

How routers work

The data is not directly sent to the internet, the computer is connected to a router that *“uses its routing table to determine the best path to use to forward the packets”* [3]. Then, the router is connected to your ISP which is finally connected to the internet. Every router has its own public IP address, and all the devices connected to it have their own private IP address, so when you make a request, the internet sees that the request is coming from a public IP address, and it will send the data to it (the router).

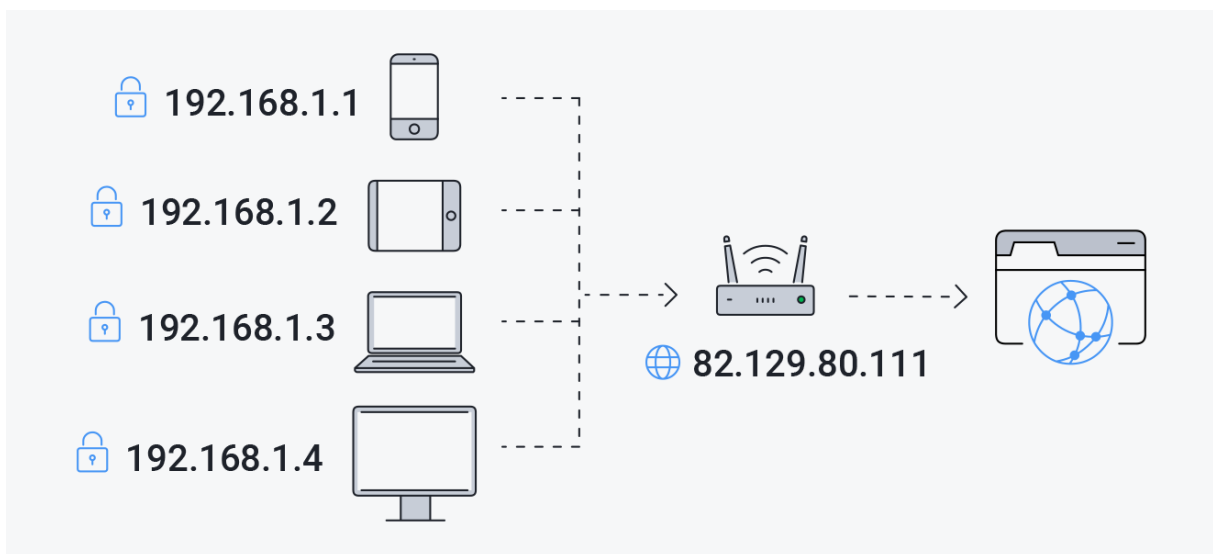


Figure 6 : public and private IP addresses

How switches work

There is something between devices and routers. Inside a LAN (Local Area Network), switches connect multiple devices on the same network [3] and allow them to communicate. Then the router connects the LAN to the Internet. When a device in a LAN makes a request, it first goes through a switch to reach the router, which then forwards the request to the website's server via the internet.

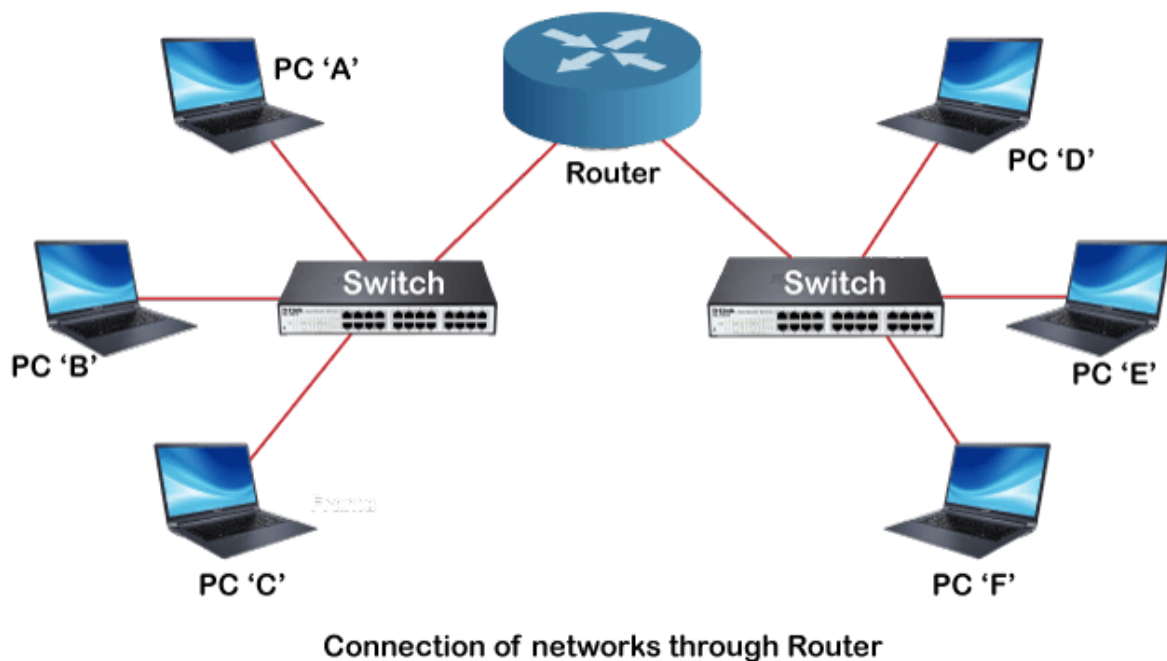


Figure 7 : Switches making the link between devices and the router

Research on routing

Introduction

Basically, routing is the process that allows the flow of data from the source to its destination. It is ruled by some protocols to make it efficient and optimized so data can flow faster. There are two types of routing, static routing and dynamic routing. *“Static routes must be manually reconfigured anytime the network topology changes” [4]*, meaning that the route for a data to follow never changes unless you change it manually. Dynamic routing uses protocols to find the best route for data to follow based on different metrics like hop count, bandwidth, delay, reliability, etc.

There are two types of dynamic routing protocols:

Type	Examples	Used for
IGP (Internal Gateway Protocol)	RIP, OSPF, EIGRP	Internal networks (LAN, WAN)
EGP (External Gateway Protocol)	BGP	Global internet

Internal Gateway Protocol (IGP)

One of the most used protocols among IGPs is OSPF (Open Shortest Path First). Routers using OSPF exchange information about their links with neighboring routers. Each router then builds a database of the network so that every router knows the shortest path to each destination.

OSPF organizes networks into areas to improve scalability. An area is a group of routers that are physically next to each other (considering a certain distance), and all these routers have the same link-state information. The backbone area (Area 0) is the core of the OSPF network, and all other areas must be connected to it. This structure reduces the amount of link-state information that routers need to maintain, limiting the number of necessary updates, improving the efficiency and rapidity in large networks.

OSPF Hierarchy

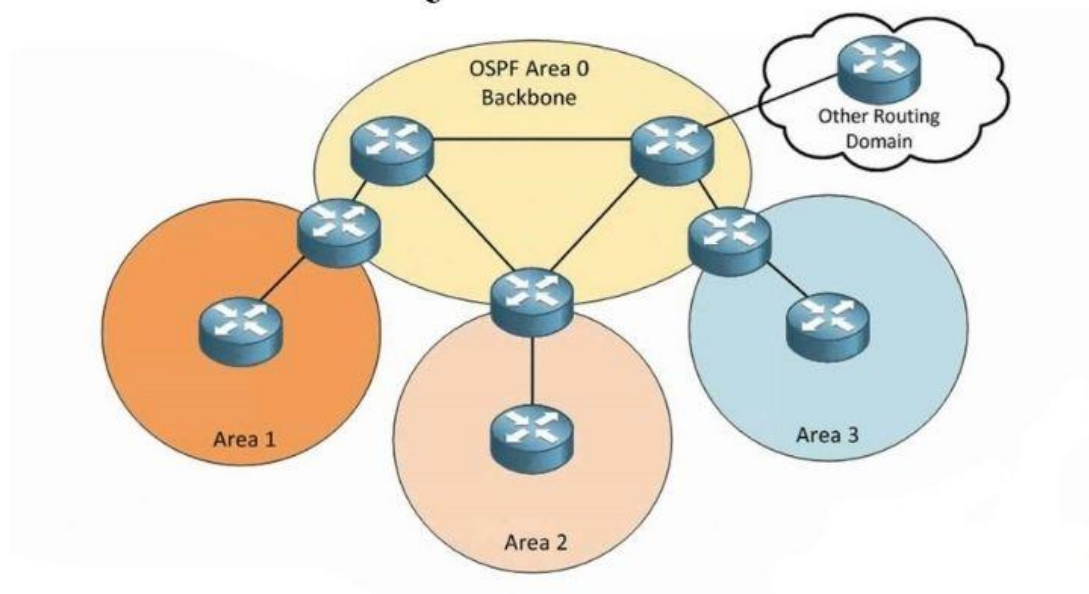


Figure 9 : Organization of OSPF networks

External Gateway Protocols (EGP)

The only EGP used on the internet is the Border Gateway Protocol (BGP). Let's imagine two OSPF areas inside the same [AS](#): Area 1 contains Network 1 & 2, and Area 2 contains Network 3 & 4. Each area manages internal communication with OSPF, and communication between areas happens through an Area Border Router (ABR, router that connects multiple areas together). If this AS needs to communicate with another AS, it will use BGP.

```

Microsoft Windows [version 10.0.22631.4830]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\ewenn>tracert www.google.com

Détermination de l'itinéraire vers www.google.com [142.250.200.4]
avec un maximum de 30 sauts :

  1    2 ms    2 ms    2 ms    10.172.128.1
  2    9 ms    7 ms    7 ms    gw-vlan45.e2.thn.ask4.net [213.143.6.129]
  3    8 ms    8 ms   58 ms    core2.lond1.uk.ask4.net [213.143.0.136]
  4   10 ms    8 ms    6 ms    core1.lond1.uk.ask4.net [213.143.0.128]
  5     *      *     10 ms    72.14.197.250
  6   12 ms   10 ms   11 ms    216.239.54.127
  7    9 ms    8 ms    8 ms    108.170.234.231
  8   10 ms    9 ms   20 ms    lhr48s29-in-f4.1e100.net [142.250.200.4]

Itinéraire déterminé.
  
```

Figure 10 : Path to "www.google.com" using BGP by typing "tracert + website's URL"

As shown in the image above, by typing "tracert www.google.com" you can see by which routers your data has travelled to reach Google's servers. In this case, the data has travelled through multiple AS before reaching destination. First, the data travelled in my ISP Autonomous System (ask4) before reaching the AS of google, this is where BGP has been used, to establish the communication between these two different AS.

Research on network services

Virtual Private network (VPN)

With the growing use of the internet, a great number of tools and services are arriving on the market to give users the best possible experience. Among them, there is VPN (Virtual Private Network) well known for allowing you to watch series that are not available in your country, but it is more than that.

A VPN securely and anonymously connects the user's device to a server. Without a VPN, the user connects directly to the internet, but with a VPN, the user is first connected to a VPN client that sends the data to the VPN server through an encrypted connection known as tunnel. Then, the VPN server sends the data to the internet under the VPN server's IP address.

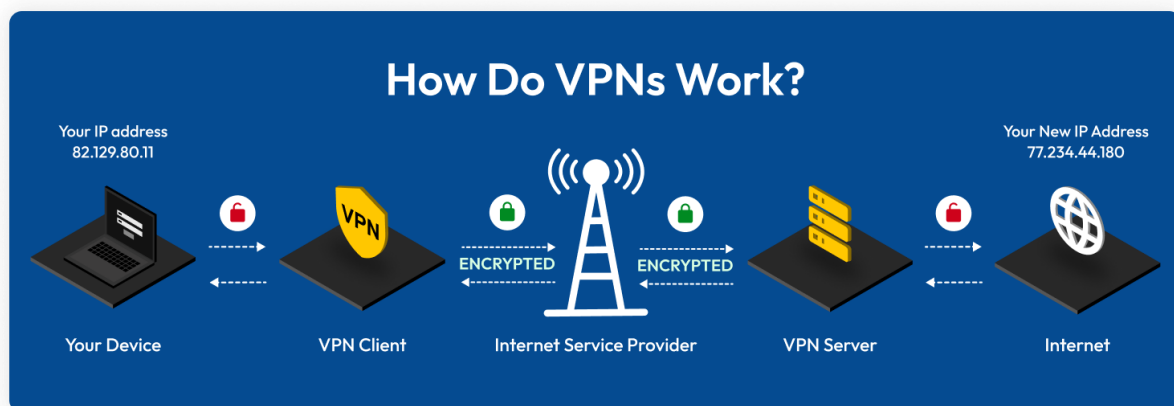


Figure 11 : How VPNs work

That way, the user private IP address is not shown on the public internet.

Active Directory Domain Services (ADDS)

Active Directory Domain Service allows you to create a domain where you can put multiple servers or clients. You can manage all these clients at the same time by applying a rule to this domain. If you create a Subdomain, you can apply a rule to it that will not be applied to the main domain.

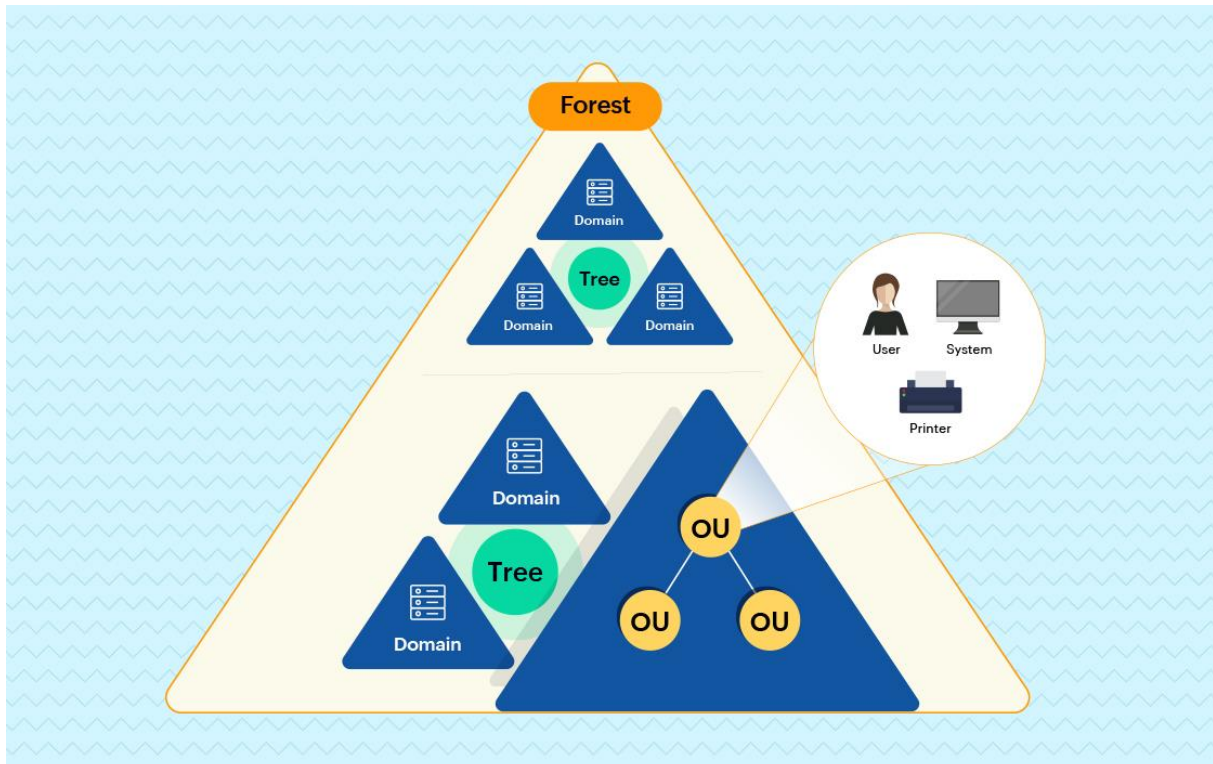


Figure 12 : Representation of Active Domain Directory Services (ADDS)

Dynamic Host Configuration Protocol (DHCP)

DHCP is a protocol that automatically gives an IP to any device on a network, it is a client/server protocol. Each device on the network will run a DHCP client to be able to ask for an IP, then, somewhere on the network, a DHCP server will reply by providing the device a unique IP address (at home your DHCP server is hosted by your router).

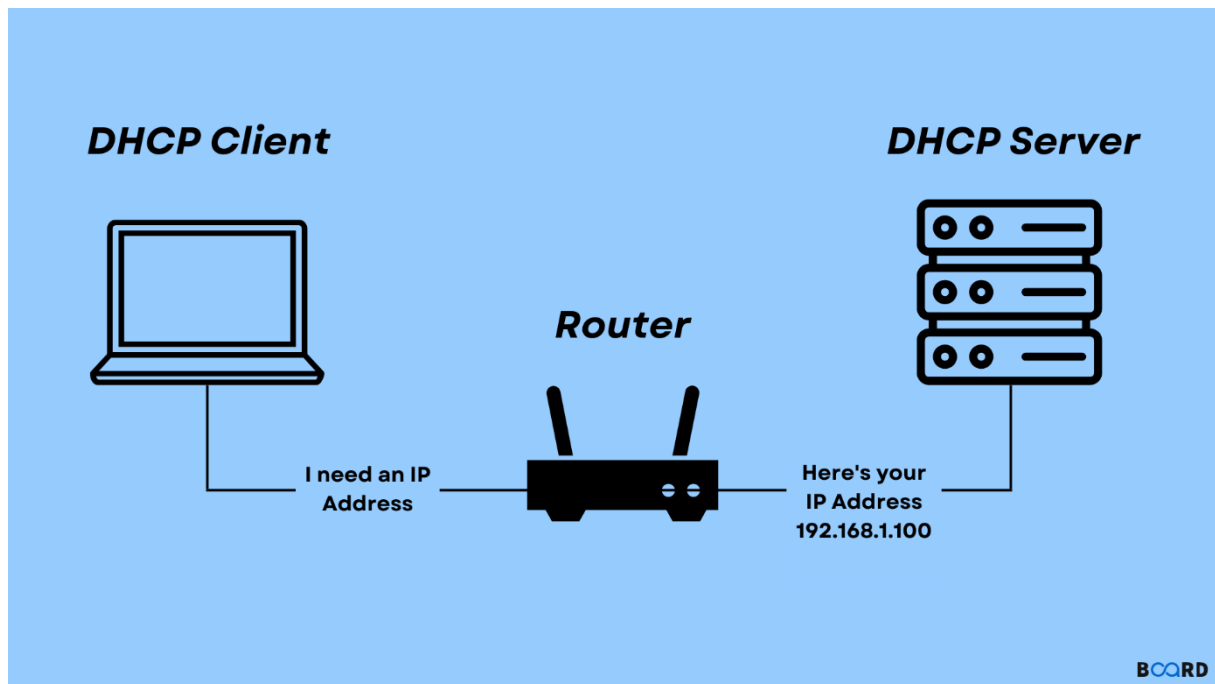


Figure 13 : How DHCP works

When the DHCP server gives an IP address, it gives it a lease time, meaning that after a certain amount of time, the device will have to renew its IP address because the DHCP server is going to take it back and put it in the list of available IP addresses.

TASK 2 | Technical Documentation

2.1 Choosing the Hypervisor

Choosing the type of hypervisor:

Because I am a beginner, a type 2 hypervisor was easier for me to install on my computer. It is also easier to understand and to use because it doesn't require a server to set it up, you just need a computer that has the minimum hardware requirements and install and run the software without complex configuration. Because they only require one computer, type 2 hypervisors are more flexible and manageable. If you make a mistake while using a type 2 hypervisor, you can delete your virtual machine very easily and start over as many times as you want because you have direct control over everything, and if you make backups every time you make a change in the virtual machine, as soon as something goes wrong you can just load the backup and restart from it without losing everything. Then, with all the details listed above, we can say that type 2 hypervisors are better for testing purposes because every feature and everything that belongs to the hypervisor is easily reachable and manageable. Finally, unlike type 1 hypervisors, many type 2 hypervisors are free to use. I think that type 2 hypervisors are the best introduction in the world of virtual machines and networks configurations.

Choosing the hypervisor:

Among all the type 2 hypervisors available in the market, I chose to use Oracle Virtual Box for many reasons. First, it is free, making it a very accessible choice for a first experience with virtual machines. Also, Oracle Virtual Box allows the user to save the state of the virtual machine at any time so if something goes wrong during the configuration or testing of the network, you can easily load the save and restart from a clean state. Oracle Virtual Box is cross-platform compatible, it can be a great feature to have depending on what operating system I am working on. Then, I am beginning with Oracle VM Virtual Box in computer forensics, so I think it is a great choice to use the same hypervisor in Network Operating Systems and Services to know very well about it at the end of the semester instead of using another hypervisor and know a little about 2 different hypervisors.

2.2 Download and Setup

Downloading the chosen Hypervisor (Oracle VirtualBox) from the official website :

<https://www.oracle.com/uk/virtualization/technologies/vm/downloads/virtualbox-downloads.html#vbox> :

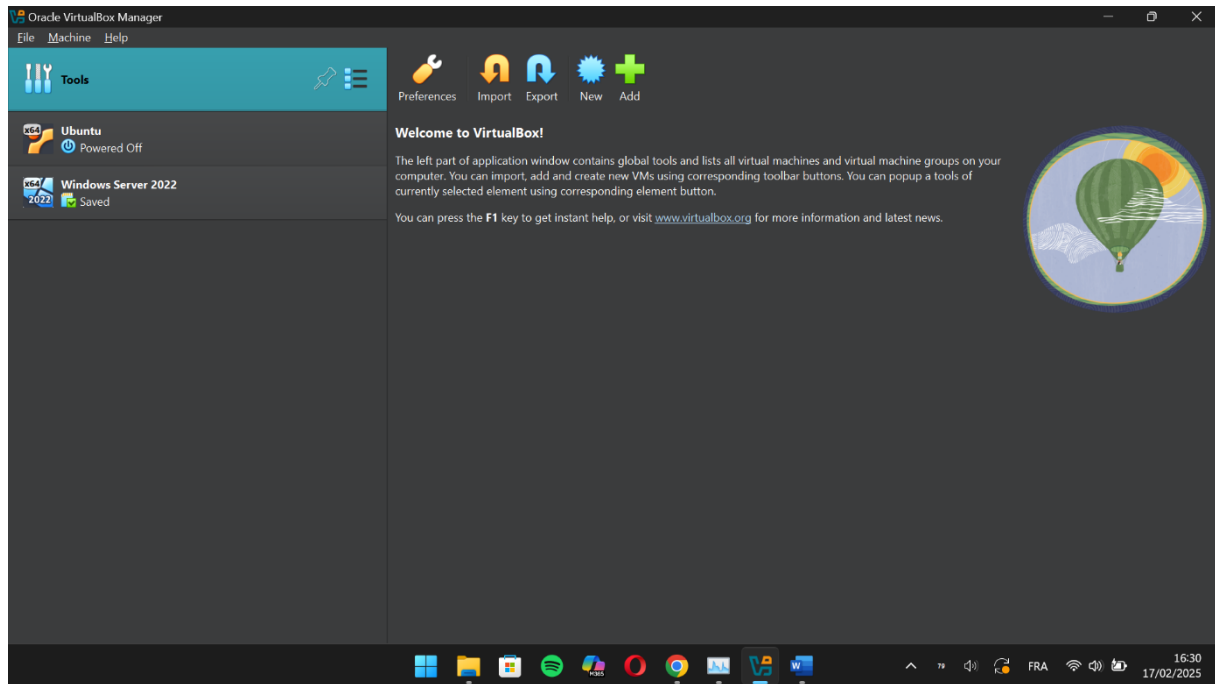


Figure 14 : Oracle Virtual Box main menu

Downloading Windows Server 2022 (S-DC) from Microsoft official website:

<https://www.microsoft.com/en-us/evalcenter/download-windows-server-2022>

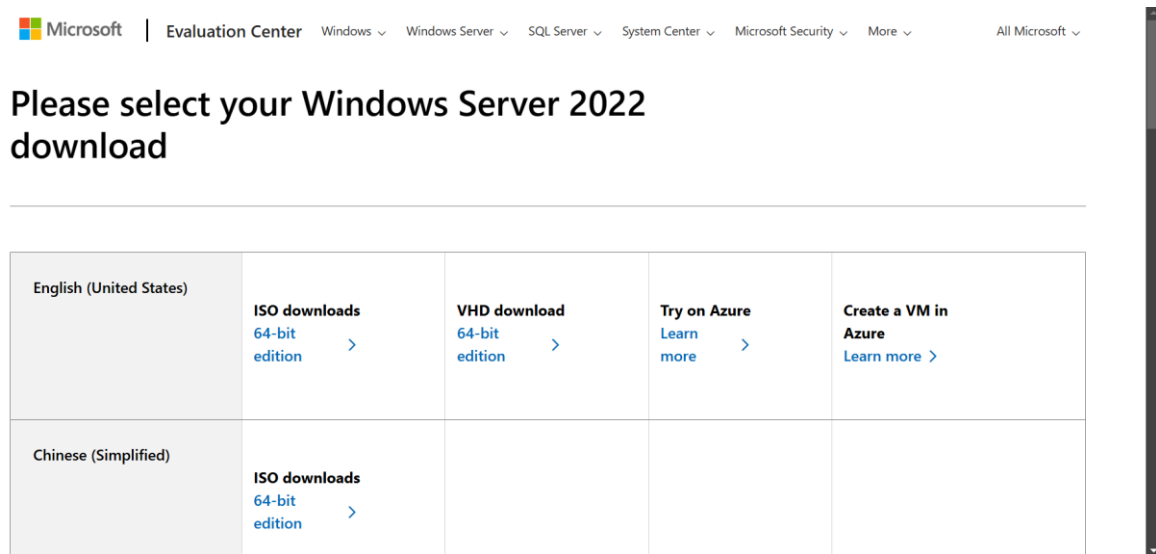


Figure 15 : Downloading the Windows Server 2022 ISO

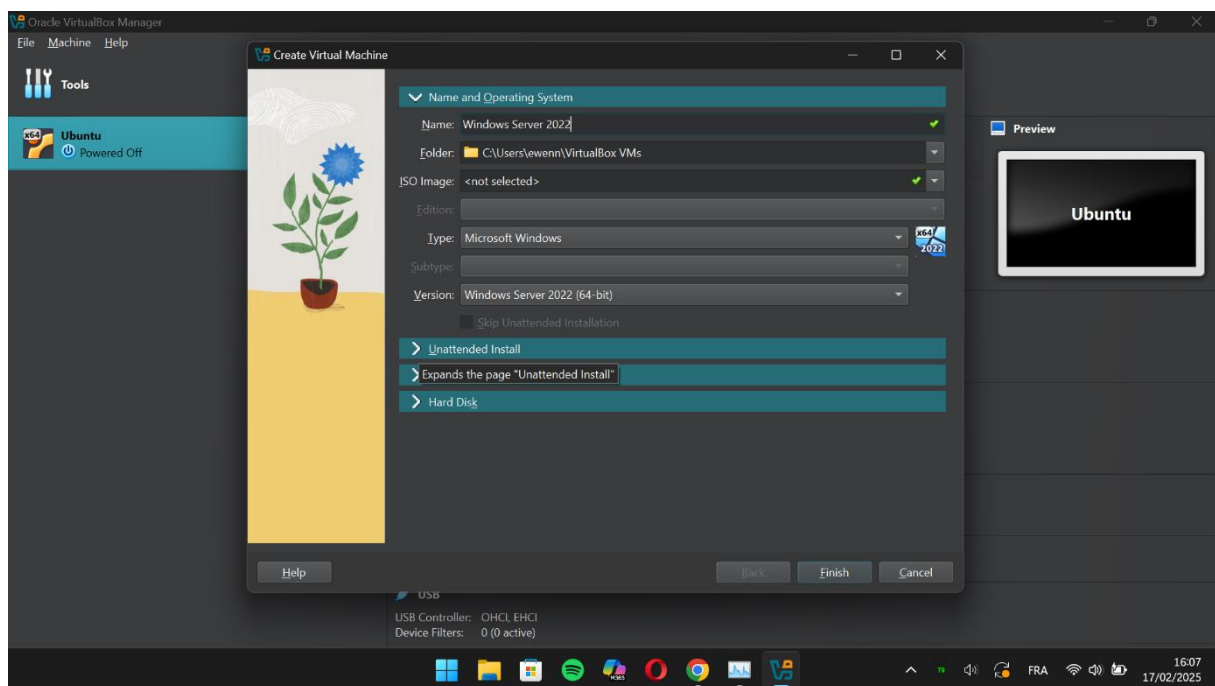


Figure 16 : Creating the Windows Server 2022 Virtual Machine

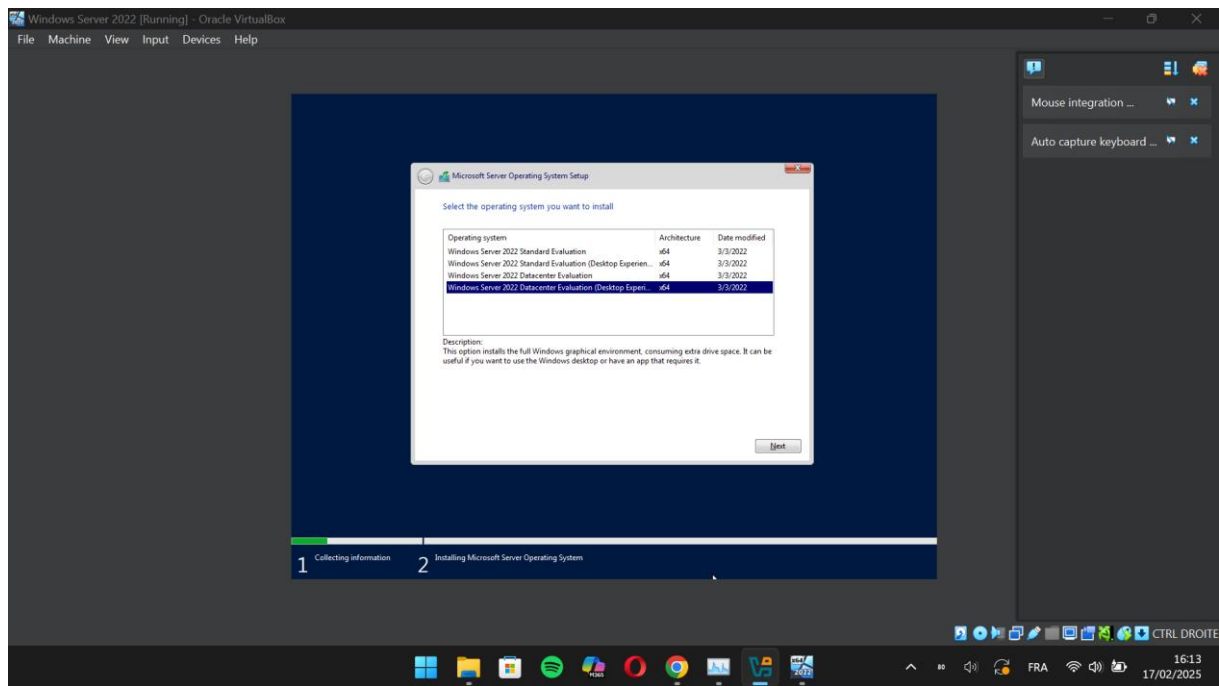


Figure 17 : Selecting the Operating System

As asked in the coursework, I installed the “Windows Server 2022 Datacenter Evaluation (Desktop Experience)” Operating System.

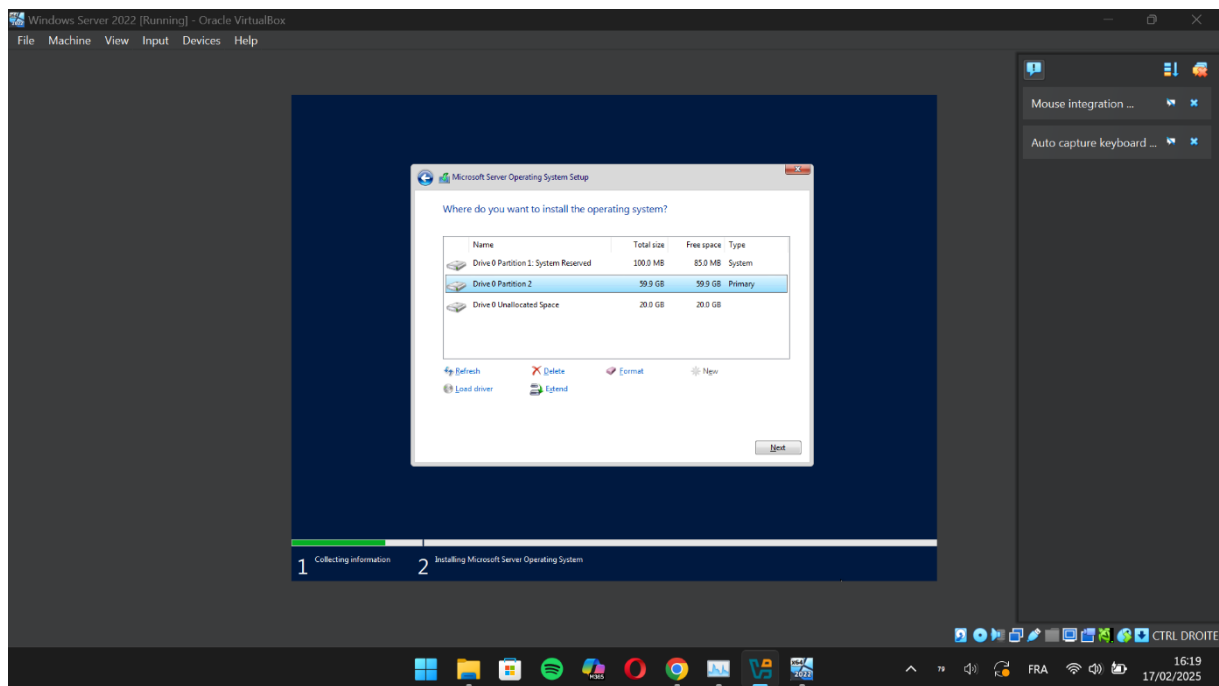


Figure 18 : Configuring storage locations

I separated the 80GB of storage allocated to the Virtual Machine into two different sized disks so I can use one for a drive or something else that could be asked for in coursework 2.

Then did the same thing for the client, I installed windows 10 (ClientS1) and created a second 5GB disk.

Downloading Debian Linux (C-Server) from Debian official website:

<https://www.debian.org/distrib/>

To complete the given topology, we need to setup the other network by downloading the Linux Server and Client.

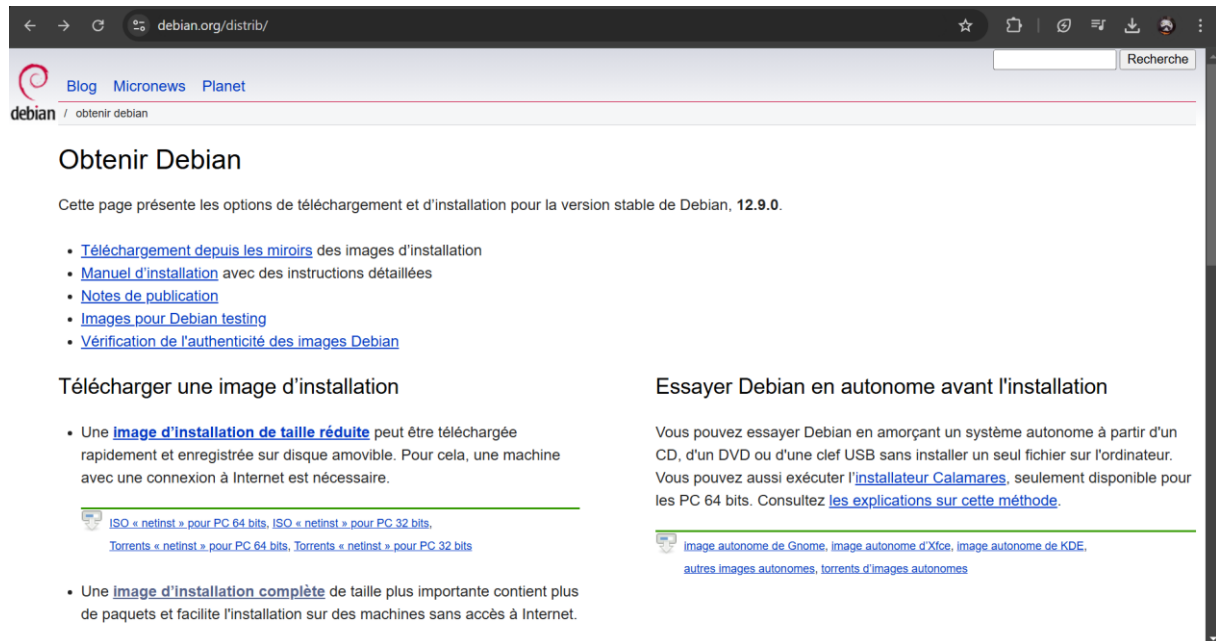


Figure 19 : Download Linux from official Debian website

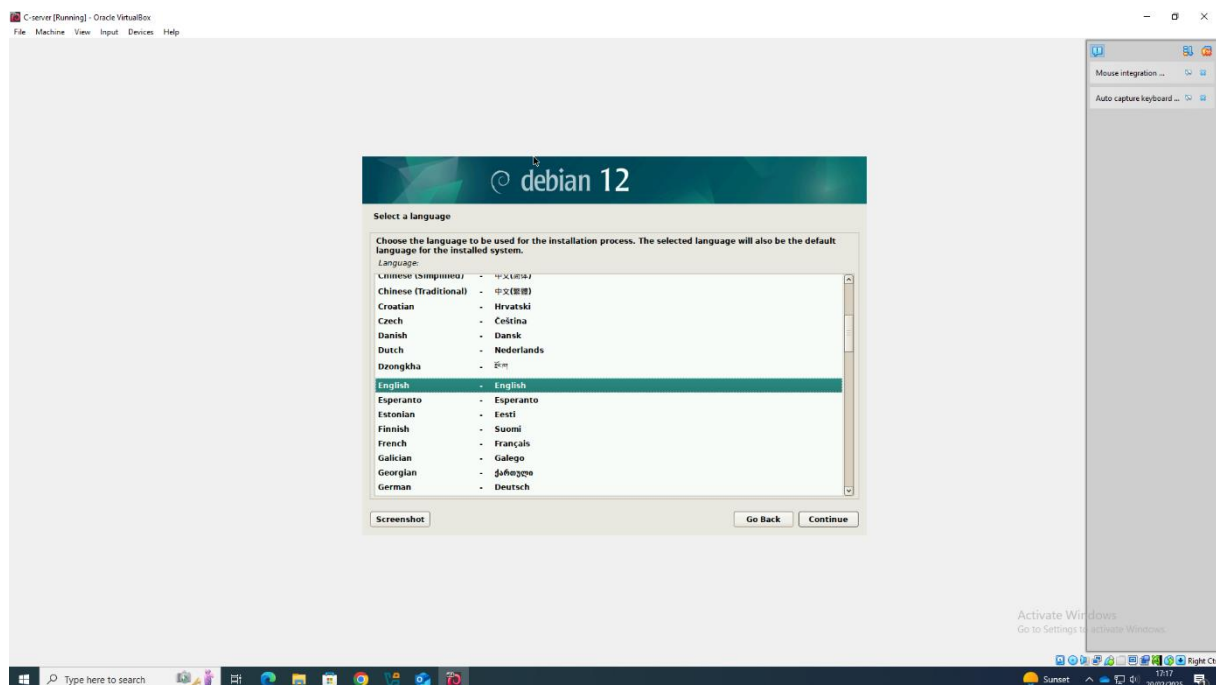


Figure 20 : Debian Linux setup

Downloading Ubuntu (ClientC1) from Ubuntu official website:

<https://ubuntu.com/download/desktop>

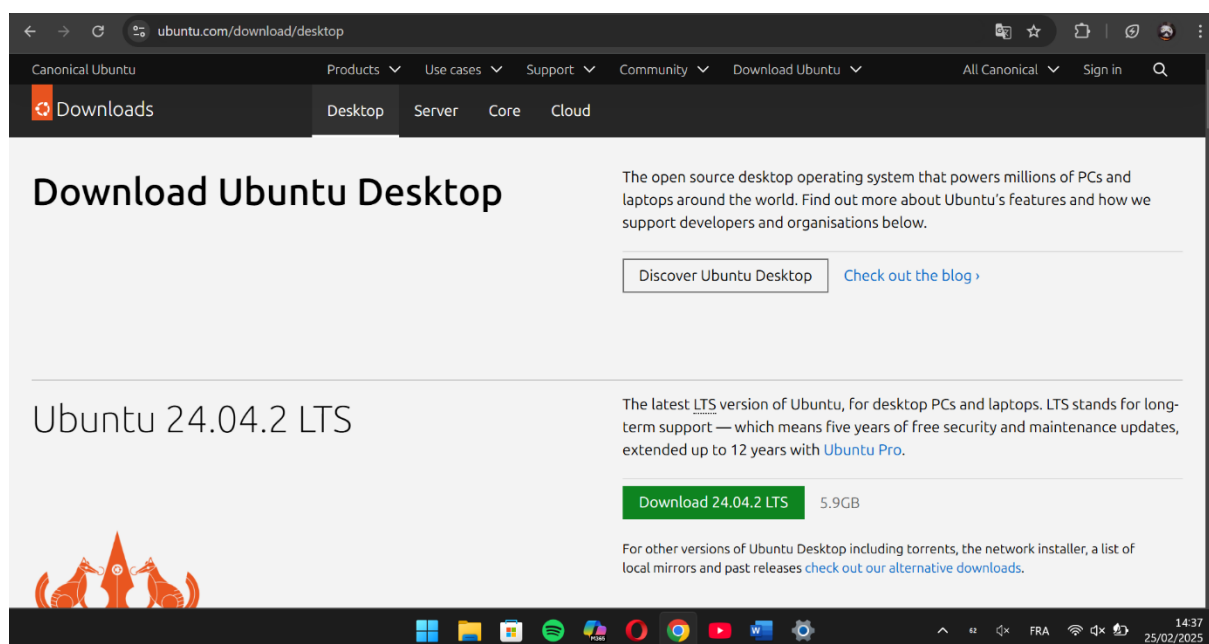


Figure 21 : Downloading Ubuntu from official Ubuntu website

2.3 Configure IP Addressing

Windows:

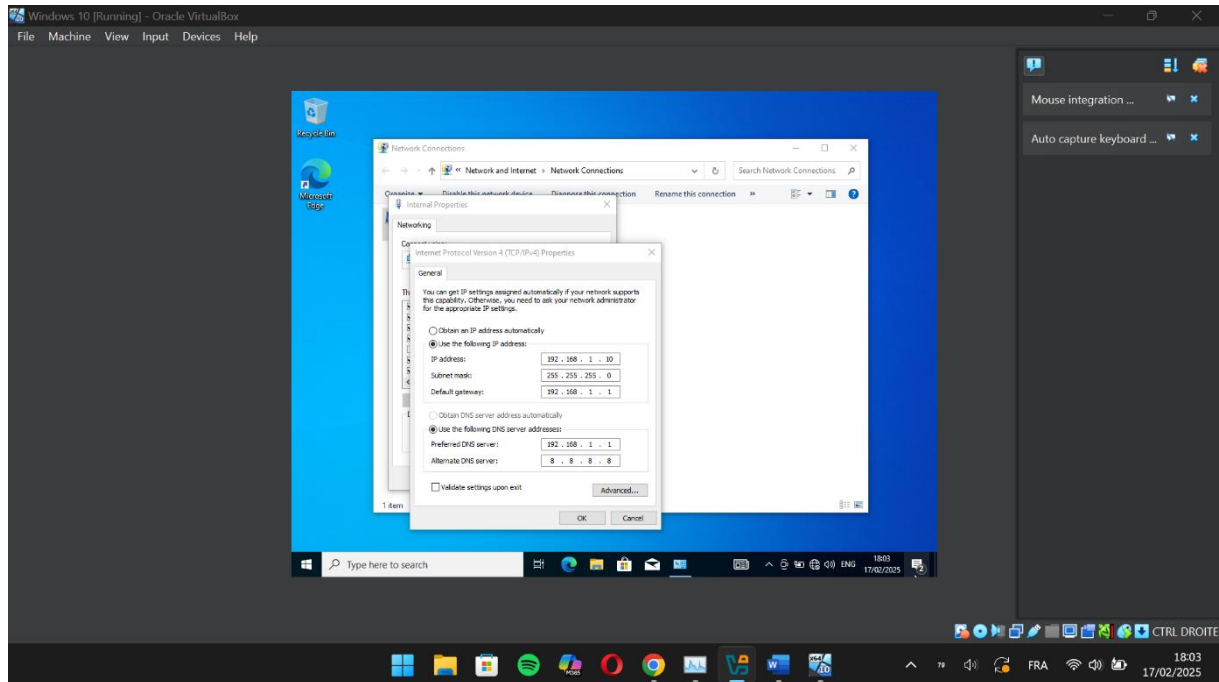


Figure 22 : Configuring the Client IP Addressing (Windows 10)

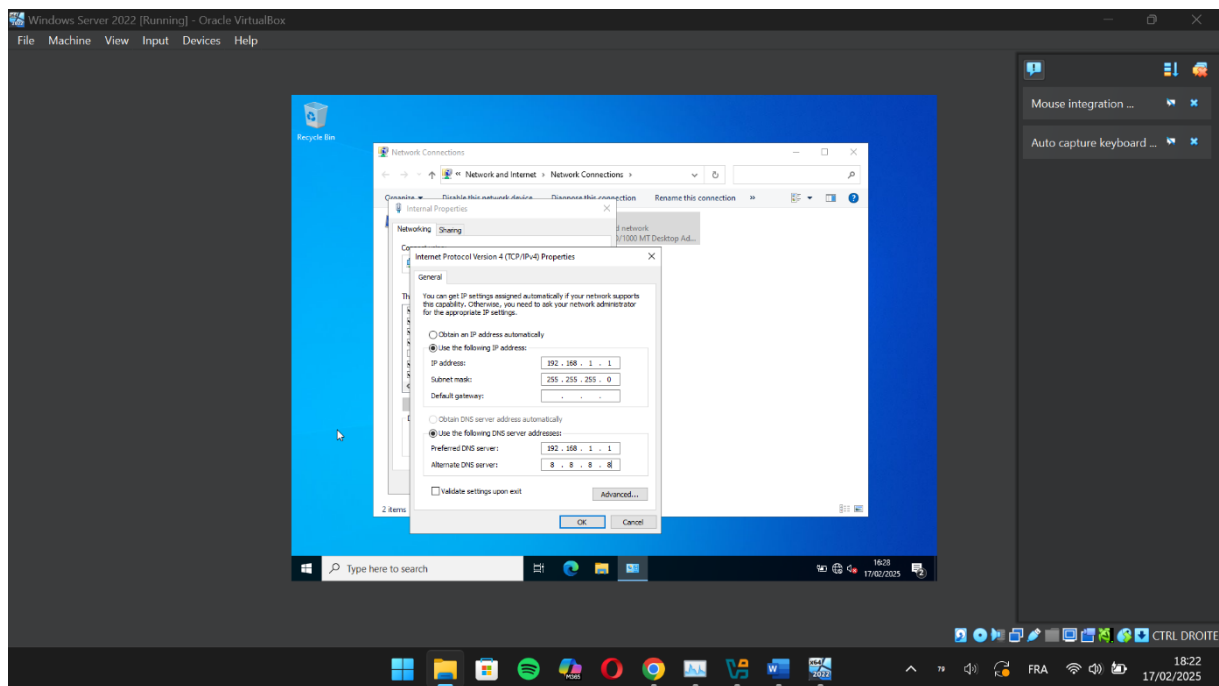


Figure 23 : Configuring the server IP Addressing (Windows Server 2022)

Linux:

After [sudo](#) is installed, we now have the rights to modify the “*interfaces*” file to configure the IP addressing of our Debian Linux server.



```

GNU nano 7.2 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# This is for host only adapters
allow-hotplug enp0s10
iface enp0s10 inet dhcp

#This is for NAT
allow-hotplug enp0s9
iface enp0s9 inet dhcp

#This is the external network connected to S-DC
allow-hotplug enp0s3
iface enp0s3 inet static
address 10.0.1.2
netmask 255.255.255.252

#This is the internal network connected to linux client
allow-hotplug enp0s8
iface enp0s8 inet static
address 192.168.1.11
netmask 255.255.255.0
  
```

Figure 24 : Configuring the IP addressing in the “*interfaces*” file, Linux Server (Debian Linux)

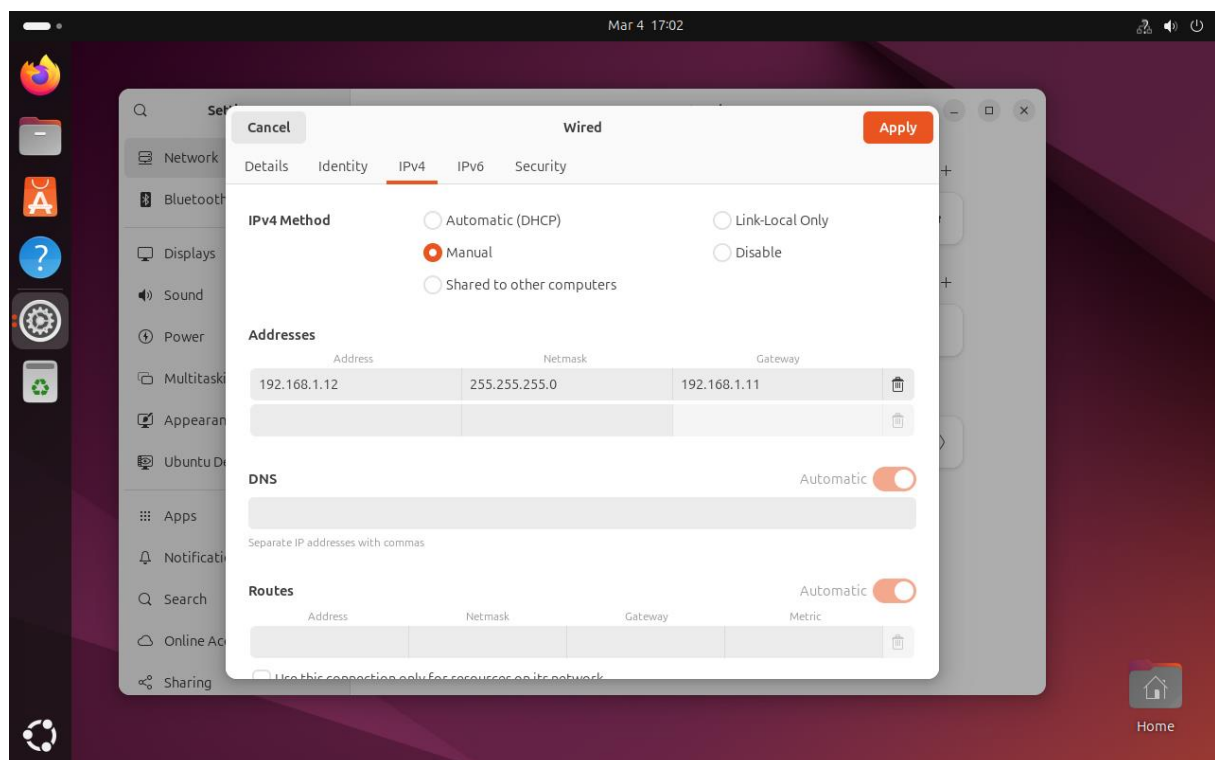


Figure 25 : Configuring the IP addressing of the Linux Client (Ubuntu)

2.4 Testing and Troubleshooting

Windows ↔ Windows:

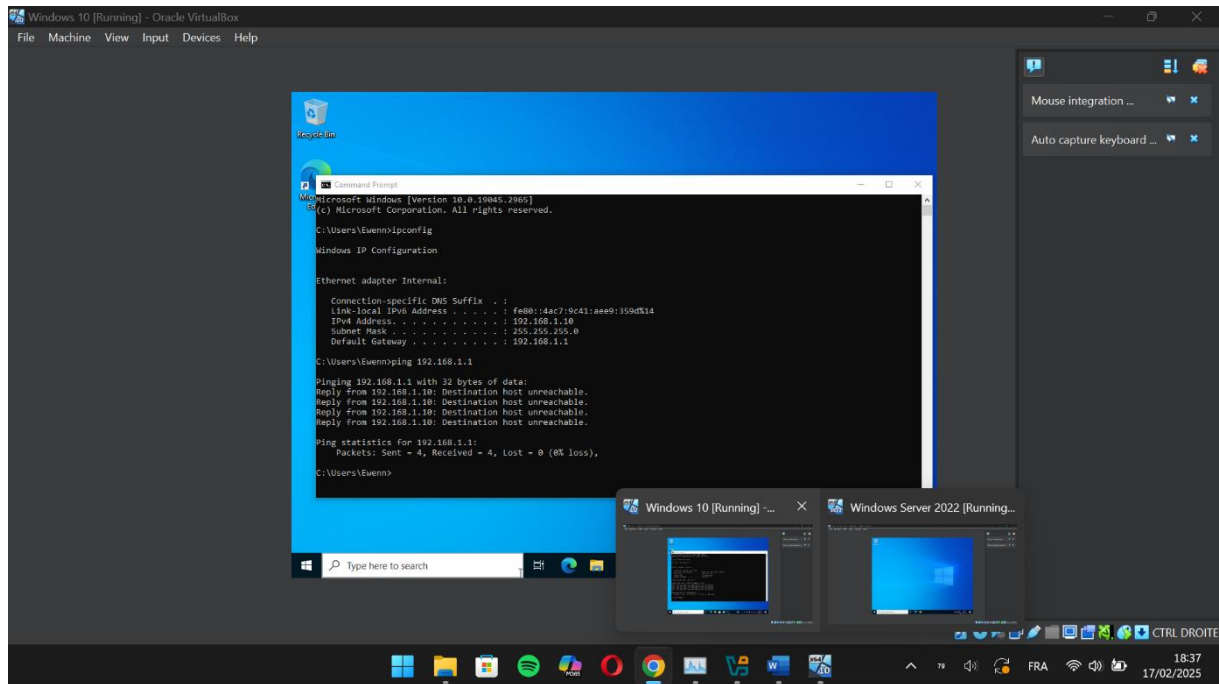


Figure 26 : Pinging the Server (S-DC) from the Client (ClientS1) (not working)

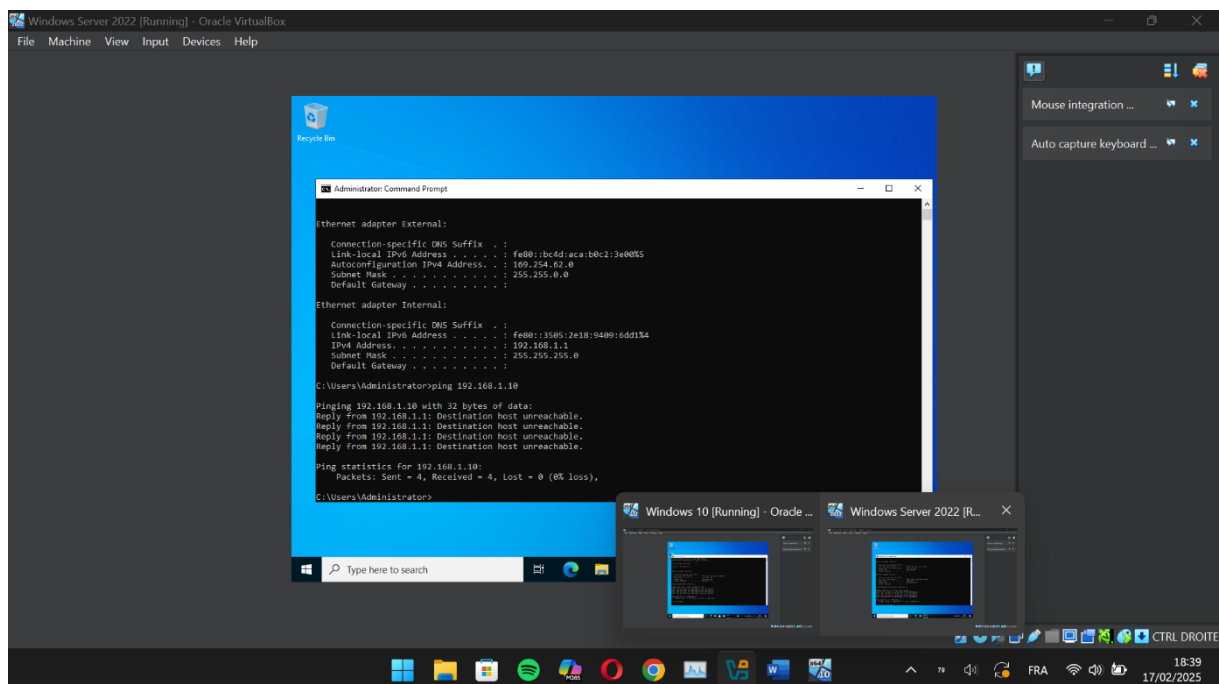


Figure 27 : Pinging the Client (ClientS1) from the Server (S-DC) (not working)

As you can see from the two images above, the ping is not successful, we are going to troubleshoot it by disabling the firewall.

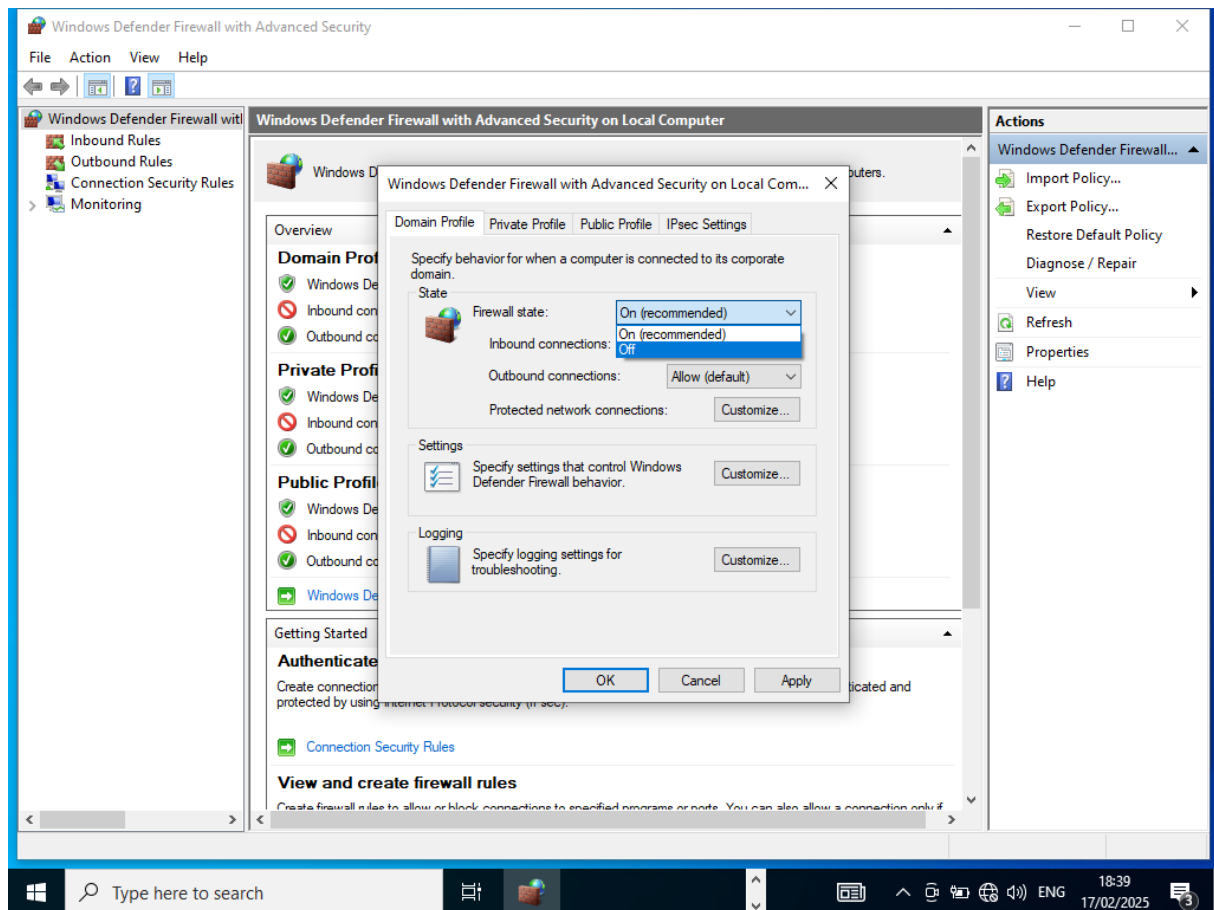


Figure 28 : Disabling all the Client's firewall profiles

We are disabling the firewall because it is the reason why the ping is not working. When you ping a device, the device you are using to ping sends ICMP packets (Internet Message Control Protocol) to the other device. One of the most common use of ICMP is to determine if data is getting to its destination and at the right time, this makes ICMP an important aspect of error reporting. In our case, the firewall doesn't allow ICMP packets, so we can't ping the client. After disabling all the client's firewall profiles, we should be able to ping the client from the server because the client will now accept the ICMP packets.

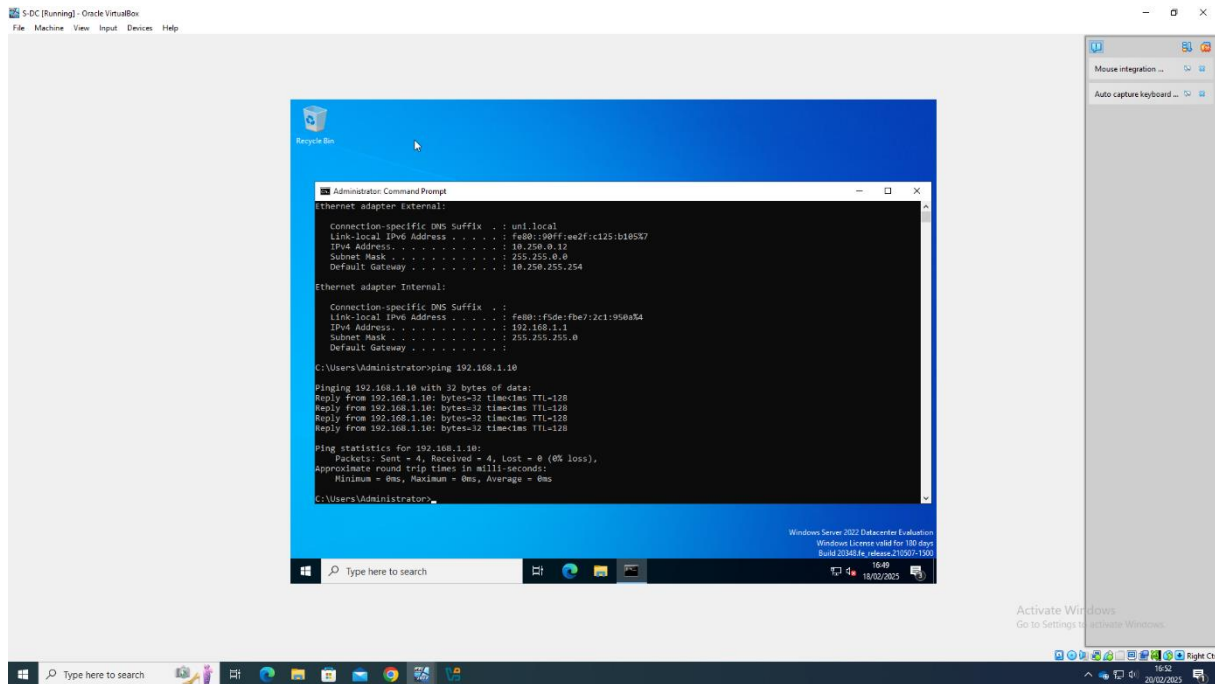


Figure 29 : Pinging the Client (ClientS1) from the server (S-DC) (working)

The ping is now successful because the Client accepts ICMP packets

Now we can ping the Client from the Server. To solve the problem of the unsuccessful ping from the Client to the Server, we will use a different method. Instead of completely disabling all the firewall profiles, we will go in the firewall's configuration and enable the Server to receive ICMP packets.

To enable the ICMP packets we need to follow this path:

- Windows Defender Firewall > Advanced settings > Inbound Rules
- Then find these two rules and ensure they are enabled:
 - o "File and Printer Sharing (Echo Request - ICMPv4-In)"
 - o "File and Printer Sharing (Echo Request - ICMPv6-In)"

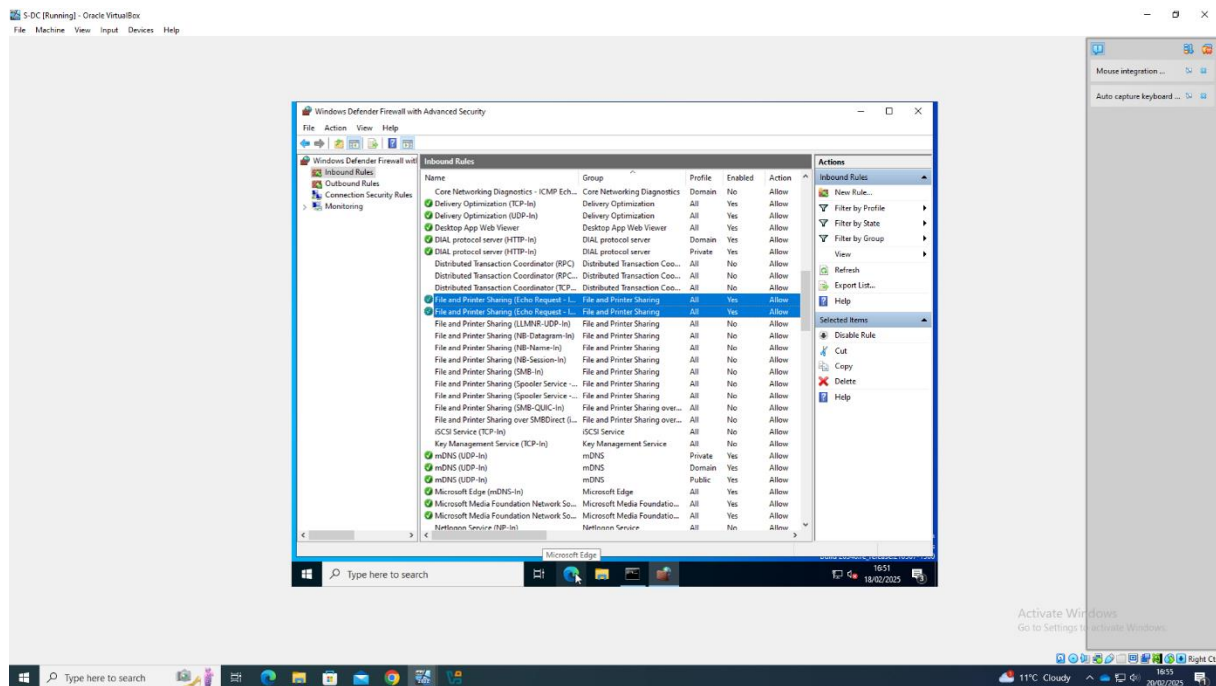


Figure 30 : Allowing ICMP packets on the Server

Now that the server accepts ICMP packets, the ping from the Client to the Server should now be successful.

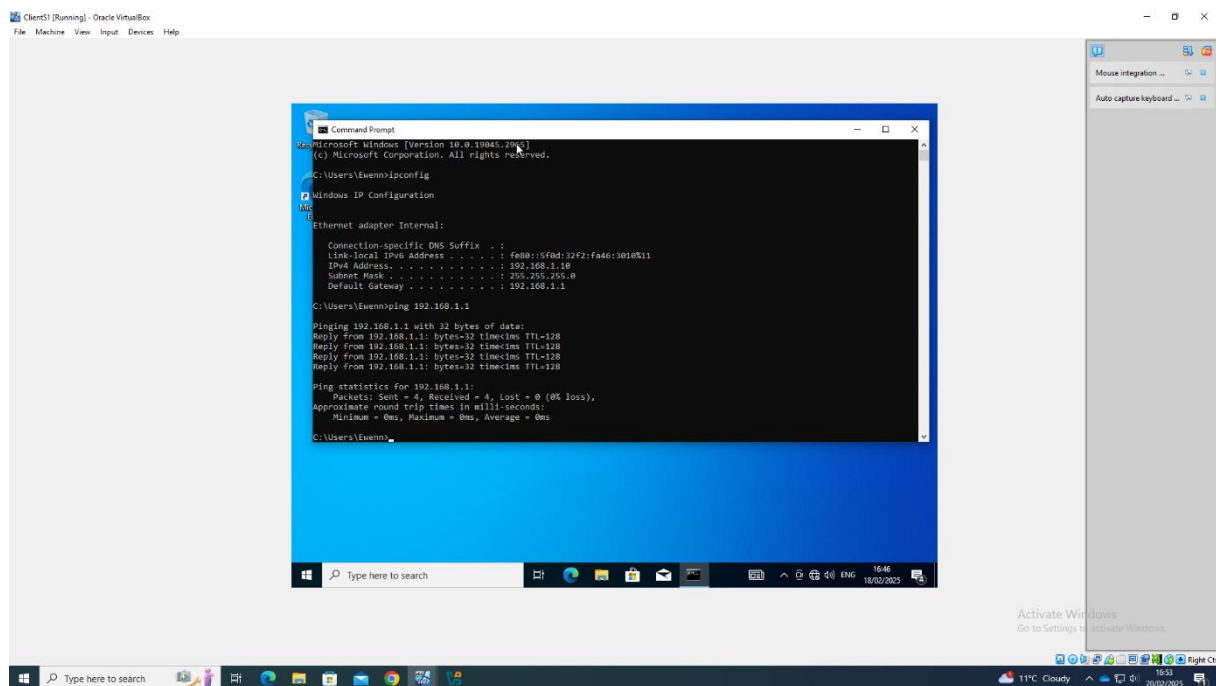
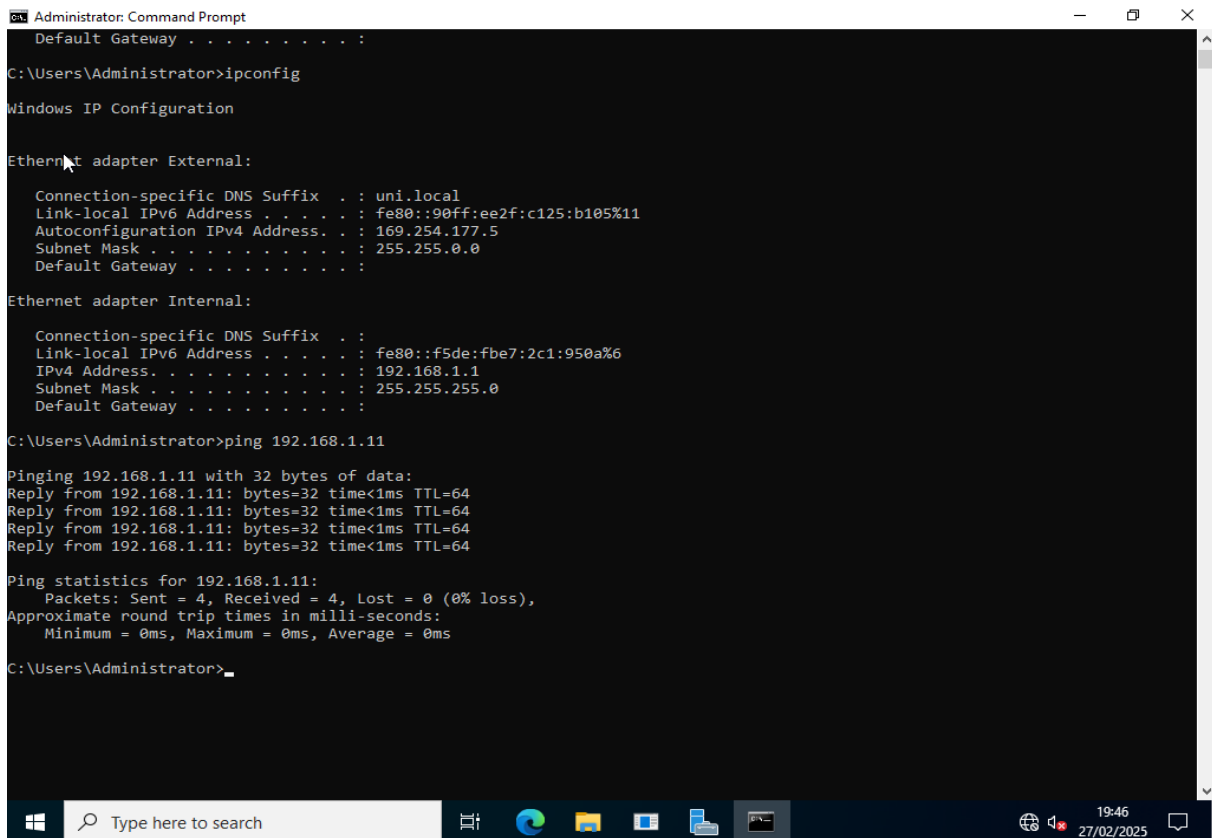


Figure 31 : Pinging the Server (S-DC) from the Client (ClientS1) (working)

As expected, we can now ping the Server from the Client.

Windows ↔ Linux:


```

Administrator: Command Prompt
Default Gateway . . . . . :
C:\Users\Administrator>ipconfig

Windows IP Configuration

Ethernet adapter External:

    Connection-specific DNS Suffix  . : uni.local
    Link-local IPv6 Address . . . . . : fe80::90ff:ee2f:c125:b105%11
    Autoconfiguration IPv4 Address. . : 169.254.177.5
    Subnet Mask . . . . . : 255.255.0.0
    Default Gateway . . . . . :

Ethernet adapter Internal:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::f5de:fbe7:2c1:950a%6
    IPv4 Address. . . . . : 192.168.1.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :

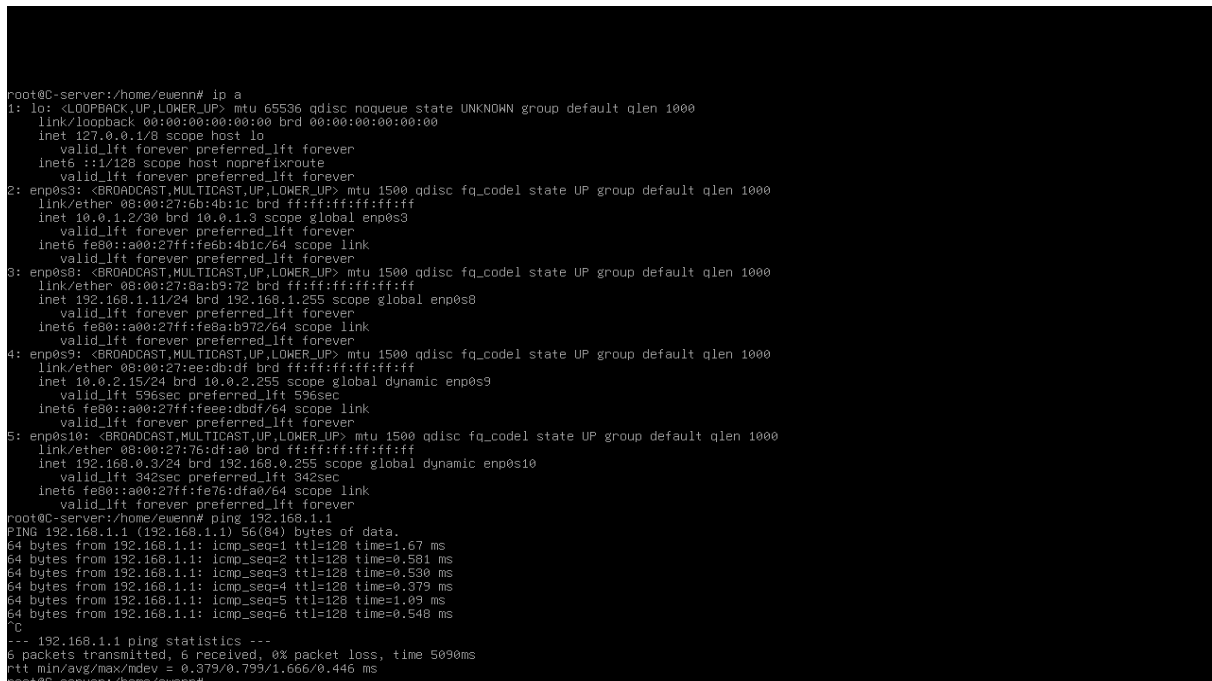
C:\Users\Administrator>ping 192.168.1.11

Pinging 192.168.1.11 with 32 bytes of data:
Reply from 192.168.1.11: bytes=32 time<1ms TTL=64
Reply from 192.168.1.11: bytes=32 time<1ms TTL=64
Reply from 192.168.1.11: bytes=32 time<1ms TTL=64
Reply from 192.168.1.11: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.1.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Administrator>
  
```

Figure 32 : Pinging Debian (C-Server) from Windows (S-DC)



```

root@C-server:/home/ewenn# ip a
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 noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:6b:4b:1c brd ff:ff:ff:ff:ff:ff
    inet 10.0.1.2/30 brd 10.0.1.3 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe6b:4b1c/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:8a:b9:72 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.11/24 brd 192.168.1.255 scope global enp0s8
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe8a:b972/64 scope link
        valid_lft forever preferred_lft forever
4: enp0s9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:ee:db:df brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s9
        valid_lft 596sec preferred_lft 596sec
    inet6 fe80::a00:27ff:feee:dbdf/64 scope link
        valid_lft forever preferred_lft forever
5: enp0s10: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:76:df:a0 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.3/24 brd 192.168.0.255 scope global dynamic enp0s10
        valid_lft 342sec preferred_lft 342sec
    inet6 fe80::a00:27ff:fe76:dfa0/64 scope link
        valid_lft forever preferred_lft forever
root@C-server:/home/ewenn# ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data:
64 bytes from 192.168.1.1: icmp_seq=1 ttl=128 time=1.67 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=128 time=0.581 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=128 time=0.530 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=128 time=0.379 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=128 time=1.09 ms
64 bytes from 192.168.1.1: icmp_seq=6 ttl=128 time=0.548 ms
^C
--- 192.168.1.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5090ms
rtt min/avg/max/mdev = 0.379/0.799/1.666/0.446 ms
root@C-server:/home/ewenn#
  
```

Figure 33 : Pinging Windows (S-DC) from Linux (C-Server)

The two servers (Windows and Linux) can ping each other, meaning that the network has been configured correctly.

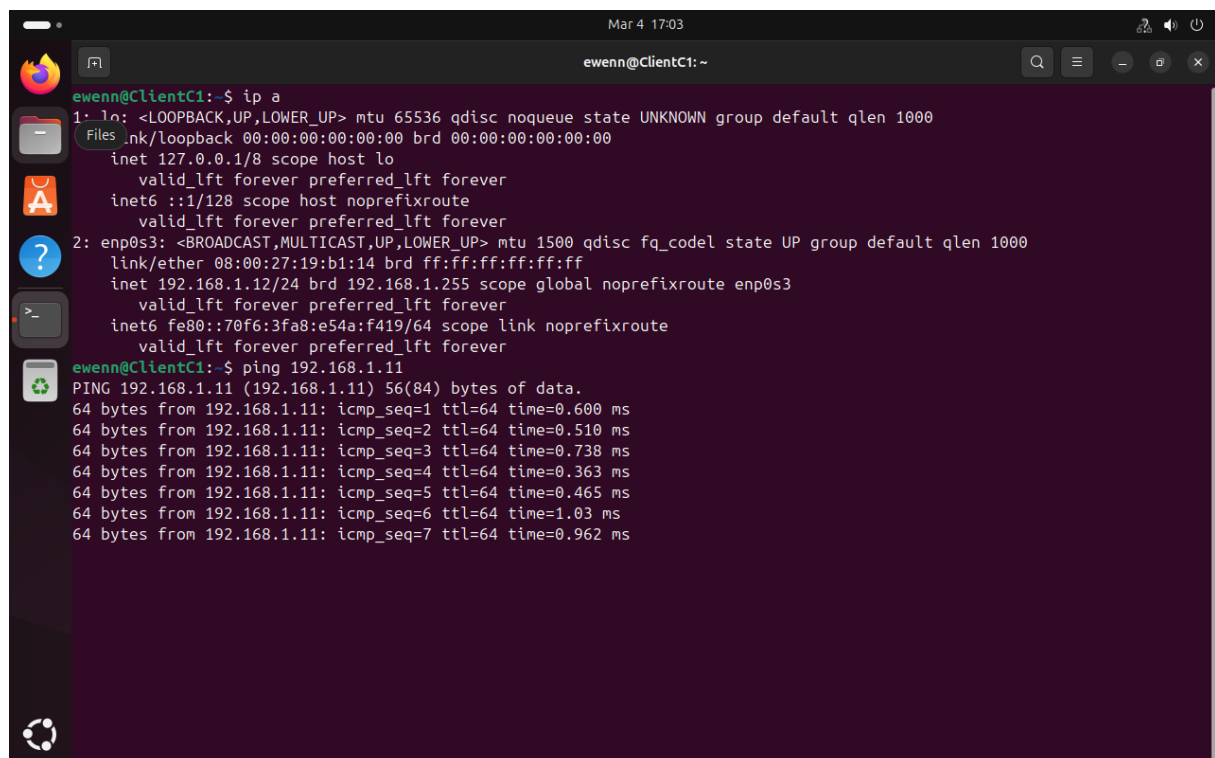
Linux ↔ Linux:

```

rtt min/avg/max/mdev = 0.306/0.803/1.951/0.411 ms
ewenn@C-server:~$ ip a
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 noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:6b:4b:1c brd ff:ff:ff:ff:ff:ff
    inet 10.0.1.2/30 brd 10.0.1.3 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe6b:4b1c/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:8a:b9:72 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.11/24 brd 192.168.1.255 scope global enp0s8
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe8a:b972/64 scope link
        valid_lft forever preferred_lft forever
4: enp0s9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:ee:db:df brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s9
        valid_lft 485sec preferred_lft 485sec
    inet6 fe80::a00:27ff:feee:dbdf/64 scope link
        valid_lft forever preferred_lft forever
5: enp0s10: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:76:df:a0 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.3/24 brd 192.168.0.255 scope global dynamic enp0s10
        valid_lft 326sec preferred_lft 326sec
    inet6 fe80::a00:27ff:fe76:dfa0/64 scope link
        valid_lft forever preferred_lft forever
ewenn@C-server:~$ ping 192.168.1.12
PING 192.168.1.12 (192.168.1.12) 56(84) bytes of data.
64 bytes from 192.168.1.12: icmp_seq=1 ttl=64 time=0.320 ms
64 bytes from 192.168.1.12: icmp_seq=2 ttl=64 time=0.749 ms
64 bytes from 192.168.1.12: icmp_seq=3 ttl=64 time=0.827 ms
64 bytes from 192.168.1.12: icmp_seq=4 ttl=64 time=1.03 ms
64 bytes from 192.168.1.12: icmp_seq=5 ttl=64 time=0.689 ms
64 bytes from 192.168.1.12: icmp_seq=6 ttl=64 time=0.965 ms
64 bytes from 192.168.1.12: icmp_seq=7 ttl=64 time=0.780 ms
64 bytes from 192.168.1.12: icmp_seq=8 ttl=64 time=0.910 ms
64 bytes from 192.168.1.12: icmp_seq=9 ttl=64 time=1.38 ms
64 bytes from 192.168.1.12: icmp_seq=10 ttl=64 time=0.726 ms
64 bytes from 192.168.1.12: icmp_seq=11 ttl=64 time=0.617 ms
64 bytes from 192.168.1.12: icmp_seq=12 ttl=64 time=0.975 ms
64 bytes from 192.168.1.12: icmp_seq=13 ttl=64 time=0.754 ms
64 bytes from 192.168.1.12: icmp_seq=14 ttl=64 time=0.841 ms
64 bytes from 192.168.1.12: icmp_seq=15 ttl=64 time=0.877 ms

```

Figure 34 : Pinging Ubuntu (ClientC1) from Debian (C-Server)



```

Mar 4 17:03
ewenn@ClientC1: ~
ewenn@ClientC1:~$ ip a
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 noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:19:b1:14 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.12/24 brd 192.168.1.255 scope global noprefixroute enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::70f6:3fa8:e54a:f419/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
ewenn@ClientC1:~$ ping 192.168.1.11
PING 192.168.1.11 (192.168.1.11) 56(84) bytes of data.
64 bytes from 192.168.1.11: icmp_seq=1 ttl=64 time=0.600 ms
64 bytes from 192.168.1.11: icmp_seq=2 ttl=64 time=0.510 ms
64 bytes from 192.168.1.11: icmp_seq=3 ttl=64 time=0.738 ms
64 bytes from 192.168.1.11: icmp_seq=4 ttl=64 time=0.363 ms
64 bytes from 192.168.1.11: icmp_seq=5 ttl=64 time=0.465 ms
64 bytes from 192.168.1.11: icmp_seq=6 ttl=64 time=1.03 ms
64 bytes from 192.168.1.11: icmp_seq=7 ttl=64 time=0.962 ms

```

Figure 35 : Pinging the Debian Linux (C-Server) from Ubuntu (ClientC1)

The Linux Server and Linux Client can communicate together.

With all the virtual machines configured and all the troubleshooting done (Windows Client ↔ Windows Server, Windows Server ↔ Linux Server, Linux server ↔ Linux Client), all the pings are successful, and we are finally meeting the Task 2 requirements. Task 2 is finally done.

TASK 3 | Reflection

3.1 Technical Reflection

Because it was my first experience with virtual machines and networks, I faced many difficulties during this coursework:

- Understanding how hypervisors work
- Understanding how some protocols work
- Setup of virtual machines
- Troubleshooting
- Working with Debian Linux (first experience with CLI interface)

3.2 Personal Reflection

With this Coursework I understood how virtualization can benefit companies and IT infrastructures. It made me understand some very important concepts used everyday in our digital world, like how my devices communicate with each other and how I have access to the internet and what protocols and services are used to grant me this access.

Conclusion

In this coursework, the installation and setup of a virtual network using VirtualBox was successfully achieved despite some challenges faced all along with the project. I think using a type 1 hypervisor was the good choice as it was understandable and manageable for a first experience and my final virtual network works well.

In the end, my virtual network met the coursework requirements, my servers can communicate together, and each client is correctly linked to his respective server and capable of communicating with them.

This project highlights the importance of network configuration and troubleshooting skills in virtual environments.

Appendix

Classes of IP addresses

Class	Used for	Number of octets designating the network portion	range
A	Large networks	The first	1.0.0.0 to 126.0.0.0
B	Medium-sized networks	The first two	128.0.0.0 to 191.255.0.0
C	Small networks	The first three	192.0.0.0 to 223.255.255.0
D	Multicast groups		224.0.0.0 to 239.255.255.255
E	Testing purposes		240.0.0.0 to 255.255.255.255

NAT

Network Address Translation is a protocol that allows multiple private IP addresses to share one single public address. It has been introduced in 1994 as a solution for the lack of IPv4 addresses in the world. As IPv4 model allows only 4.3 billion different addresses to exist, we had to find a solution because 4.3 billion addresses were not enough for the number of connected devices around the world. *“When a local computer attempts to connect to an internet resource, the NAT device makes the connection instead. Any packets received from the internet resource are translated into the address scheme of the local network and forwarded to the local computer that initiated the connection”*

Autonomous Systems

An Autonomous System (AS) is a collection of IP networks and routers under the control of a single administrative entity (such as an Internet Service Provider (ISP), a company, or a university) that follows a common routing policy.

Sudo

Sudo is a tool that gives the user privileges to execute sensitive commands like making updates, installations, configure the firewall and the network, ...

References

[1] “Mastering Windows Server 2016 Hyper-V”, John Savill [book]

<https://www.vmware.com/topics/hypervisor>

<https://www.vmware.com/topics/network-virtualization>

<https://www.softobotics.com/blogs/exploring-hypervisors-in-cloud-architecture/>

<https://www.ibm.com/think/topics/virtualization>

[2] “TCP/IP JumpStart – Internet Protocol Basics – Second Edition”, Andrew G. Blank [book]

<https://www.cloudflare.com/fr-fr/learning/network-layer/how-does-the-internet-work/>

<https://www.avg.com/en/signal/public-vs-private-ip-address>

<https://www.broadbandsearch.net/blog/how-does-the-internet-work>

[3] “Routing and Switching Essentials”, Cisco | Networking Academy [book]

[4] “Scaling Networks”, Cisco | Networking Academy [book]

[https://www.catchpoint.com/dynamic-routing-protocols/bgp-vs-](https://www.catchpoint.com/dynamic-routing-protocols/bgp-vs-ospf#:~:text=There%20are%20several%20routing%20protocols,Border%20Gateway%20Protocol%20(BGP).)

[ospf#:~:text=There%20are%20several%20routing%20protocols,Border%20Gateway%20Protocol%20\(BGP\).](https://www.catchpoint.com/dynamic-routing-protocols/bgp-vs-ospf#:~:text=There%20are%20several%20routing%20protocols,Border%20Gateway%20Protocol%20(BGP).)

[5] “SAMS Teach Yourself in 24 Hours – TCP/IP, Joe Casad [book]

[https://learn.microsoft.com/en-us/windows-](https://learn.microsoft.com/en-us/windows-server/networking/technologies/dhcp/dhcp-top)

[server/networking/technologies/dhcp/dhcp-top](https://learn.microsoft.com/en-us/windows-server/networking/technologies/dhcp/dhcp-top)

GPT-4 to understand some concepts