



Weekly Schedule

The following is the weekly course schedule for OSM620. It includes topic schedules, assessment due dates, and course work weighting.

This is a working schedule and may be subject to change.

Week 1: September 1-7

Agenda/Topic	Reading	Assessment Due
Introduction to OSM620	Lab 1 - Environment Setup	
Introduction to Microsoft Server 2025 and Virtual Machines	Posted slides and lecture material	

Week 2: September 8-14

Agenda/Topic	Reading	Assessment Due
Microsoft Technologies, Windows Server, and Editions	Lab 2 - Hyper-V, NAT, and Windows Clients	Lab 1 - Environment Setup (2.5%)

Agenda/Topic	Reading	Assessment Due
Demo: Server Installation	Posted slides and lecture material	

Week 3: September 15-21

Agenda/Topic	Reading	Assessment Due
Basic Networking and Security Principles	Lab 3 - Security and Remote Connectivity	Lab 2 - Hyper-V, NAT, and Windows Clients (2.5%)
Demo: VM Interconnectivity and Troubleshooting	Posted slides and lecture material	

Week 4: September 22-28

Agenda/Topic	Reading	Assessment Due
DNS: Concepts and Implementation	Lab 4 - Implementing DNS	Lab 3 - Security and Remote Connectivity (5%)

Agenda/Topic	Reading	Assessment Due
Demo: Hosting DNS	Posted slides and lecture material	

Week 5: September 29 - October 5

Agenda/Topic	Reading	Assessment Due
DHCP: Concepts and Implementation	Lab 5 - Implementing DHCP	Lab 4 - Implementing DNS (5%)
Demo: Hosting DHCP	Posted slides and lecture material	

Week 6: October 6-12

Agenda/Topic	Reading	Assessment Due
Release and Discuss: Assignment 1	Assignment 1	Lab 5 - Implementing DHCP (5%)
Dedicated lab time and lab help (Work Week)	Posted slides and lecture material	Quiz 4

Week 7: October 13-19

Agenda/Topic	Reading	Assessment Due
Review and Lab Help	All previous material	Midterm Test (15%)
Midterm Test		Assignment 1 (15%)

Study Week: October 20-26

No classes this week. Campus is open.

Week 8: October 27 - November 2

Agenda/Topic	Reading	Assessment Due
Introduction to Active Directory	Lab 6 - Active Directory	
Demo: AD Creation	Posted slides and lecture material	

Week 9: November 3-9

Agenda/Topic	Reading	Assessment Due
Domain Controllers and RODCs	Lab 6 - Active Directory	
Assignment 2 Discussion: Group Project Creation	Posted slides and lecture material	

Week 10: November 10-16

Agenda/Topic	Reading	Assessment Due
Active Directory Users, Groups, and Group Policies	Lab 7 – Users, Groups, and Group Policies	Lab 6 - Active Directory (5%)
Demo: Applying Group Policies and Downstream Effects	Posted slides and lecture material	

Week 11: November 17-23

Agenda/Topic	Reading	Assessment Due
Server Storage and File Shares	Lab 8 – Configuring Storage	Lab 7 – Users, Groups, and Group Policies (5%)
Demo: Assigning and Mapping Storage	Posted slides and lecture material	

Week 12: November 24-30

Agenda/Topic	Reading	Assessment Due
Introduction to Microsoft Azure and Cloud Computing	Lab 9 – Introduction to Azure	Lab 8 – Configuring Storage (5%)
Demo: Introduction to Azure UI and Access	Posted slides and lecture material	

Week 13: December 1-7

Agenda/Topic	Reading	Assessment Due
Group Project Presentations	Posted slides and lecture material	Group Presentations
		Lab 9 - Introduction to Azure (5%)
		Assignment 2: Group Project (15%)

Week 14: December 8-12

Agenda/Topic	Reading	Assessment Due
Course Review	All previous material	Final Test (15%)

Lab 1 - Environment Setup

Lab Preparations

Purpose of Lab 1

In this lab you will stand up the base environment you'll use for the course. You'll obtain official installation media and keys, prepare VMware Workstation, and deploy two servers:

- srv1: **Windows Server 2025 Datacenter, GUI** (Desktop Experience)
- srv2: **Windows Server 2025 Datacenter, CLI** (Core)

You'll complete essential post-install tasks (time zone, naming, activation, updates), then place both servers on a two-NIC layout so they can talk to each other on a private subnet while still reaching the internet for updates.

By the end, the machines are clean, consistent, and ready for later labs (Hyper-V, security hardening, DNS, DHCP, and eventually AD).

Objectives

By the end of this lab, you will be able to:

- Acquire **Windows Server 2025 Datacenter** installation media and your individual product key from Azure Education and store them securely.
- Access **VMware Workstation** on a Seneca lab PC (locally or via MyApps) or install it on a personal PC.

- Provision two VMs with the required specs and networks:
 - srv1-senecaUsername (GUI) with NAT + VMnet10 NICs.
 - srv2-senecaUsername (Core) with NAT + VMnet10 NICs.
- Complete essential post-install tasks on each server.
- Verify basic Internet connectivity prerequisites for later labs within each VM.

Here's a basic flowchart of what we'll be doing with Lab 1 to give you a visual overview:

Download Installation
Media



Prepare VMware
Workstation



Create srv1 VM



Install Windows Server
2025 on srv1



Complete Post-Install
Tasks - srv1



Minimum Requirements

Before beginning, you must have:

1. Attended the Week 1 lecture.
2. Read through the Week 1 slides, and have them handy as a reference for concepts.
3. Working access to portal.azure.com with your Seneca credentials.
4. Access to VMware Workstation (Seneca lab PC or personal PC meeting course specs).
5. Your external SSD with at least 500 GB of dedicated space for this course.
6. Your assigned **UID** (from Blackboard Grades) handy for addressing.
7. Your physically printed **OSM620 Lab Logbook** for notetaking and saving commands.
8. Optional, but recommended: Caffeine delivery system.

Investigation 1: Downloading Installation Media

In this investigation, you will be downloading your Windows OS installation media by logging into your Seneca-based Azure account. You will also generate your personal serial keys.

1. Navigate to the Microsoft Azure site: portal.azure.com
2. Use your Seneca e-mail address and password to login.
3. Once on the main Azure page, look for the Search bar at the top of the page.
4. In the Search bar, type: **Education**, then hit Enter.

5. In the **Education | Overview** page, look to the left. You will see menu items already displayed on screen. (*Overview, Learning resources, etc.*)
6. Inside **Learning resources** in the left menu, click on **Software**.
7. In the main **Software** page, there is a Search bar just below the word Software (it says *Search* inside it.)
8. In that search field, type and enter: **Windows Server 2025 Datacenter**
9. In the item that appears below (there should only be one), click the link for **Windows Server 2025 Datacenter**.
10. On the right, an information box appears describing the software. Using your mouse to hover over this information box, scroll down to the bottom.
11. You should now see two items: **View Key** and **Download**
12. Click on **Download** first to begin downloading the Server 2025 ISO. You will need this for your operating system installation. (Don't forget where you've saved it!)
13. While the ISO file is downloading, click on **View Key**.
14. Copy this key into a text file that you save locally on your personal computer or personal USB key. You will need this for the Server installation and for any reinstalls later in the semester. **Do not lose this key and do NOT share it with anyone!**
15. Repeat steps 8-14 for the **Windows 11 Education** ISO and serial key. You will need that for later.

Reminder: Always store all serial keys in a secure location only you have access to.

Investigation 2: Using VMware Workstation

The use of *VMware Workstation* is required for this course. You can either use a

Seneca Lab computer (Option 1a / Option 1b) or a personal computer (Option 2).

If you use an external SSD, you can use both!

WARNING: Seneca Lab computers erase any locally saved data when restarted from their internal drives. Save all work to an external drive or online storage.

Option 1a: Seneca Lab Computers w/VMware Workstation Installed Locally

How do you know if your particular Seneca Lab classroom has VMware Workstation installed?

When you log in, the VMware Workstation icon will be visible on the desktop. If it is not, jump to *Option 1b*.

1. Log in to the Seneca workstation with your Seneca username (don't include @myseneca.ca) and password.
2. Once the desktop has fully loaded, double-click on the **VMware Workstation** icon on the desktop. (If not present, jump to *Option 1b* instructions.)
3. When the **VMware Workstation** application appears on screen, continue to the next Investigation.

Option 1b: Seneca Lab Computers *without* VMware Workstation Installed

Do not use this option if your lab computer already has VMware Workstation installed. Go back to Option 1a.

1. Log in to the Seneca workstation with your Seneca username (don't include @myseneca.ca) and password.
2. Once the desktop has fully loaded, double-click on the **Seneca MyApps** icon on the desktop.
3. In the browser that launches, log in with your Seneca credentials.
4. Do not ignore dialog boxes! Read them carefully instead of just clicking *Cancel* or randomly.
5. In the search bar in the middle of the screen, type *VMware* and hit **Enter**.
6. One result should show up, **VMware WS Pro**. Click this result.
7. In the new *VMware WS Pro* page, click on the version number (Example: 17.6.1). This will launch the application over the network.
8. Click on allow for any dialog boxes that appear. Remember, **do not ignore dialog boxes!**
9. Be patient. It can take a bit for the background files to transfer over the network.
10. When complete, the **VMware Workstation** application should appear on screen. (You will also see the **VMware Workstation** icon on the desktop. You can launch it from here if you accidentally close the program.)
11. Proceed to the next investigation.

Option 2: Personal Computer

Part 1: Computer Hardware Requirements

To run this course and its tech on a personal computer, it must meet the following minimum system requirements:

1. CPU: **6-core Intel/AMD**
2. RAM: **32 GB**
3. Storage: **500 GB SSD free**, *reserved for this course* (if external, USB3.0 or

higher)

4. Internet connection: **High speed, stable**

CPU architecture is vital. You *cannot* use ARM or Snapdragon based computers for this course. This includes all Apple Silicon computers (M1/M2/M3/M4, etc).

Part 2: Registering Your Broadcom Account

Broadcom (formerly VMware) now requires account registration before you can download the VMware Workstation software. Follow the instructions below to do so. You only have to do this once. (If you already have an account using your Seneca e-mail, skip to Part 3.)

1. Visit the Broadcom registration site: profile.broadcom.com/web/registration
2. Provide your Seneca e-mail address and continue.
3. You will be sent a verification code. Check your Seneca e-mail and enter the code on the page.
4. On the *Complete your Registration* page, enter your information and click **Create Account**.
5. On the *Registered Successfully!* page, you can skip the rest and simply click **I'll do it later**.
6. At the top right of the screen, click on the **Login** button. (No, you aren't logged in automatically. Yes, it's annoying.)
7. Enter your Seneca e-mail address as your username and the password you chose on Part 4 as your credentials.
8. When complete, you should see your name in the top right of the screen instead of *Login*. If not, attempt to log in again or ask for help.

Part 3: Downloading VMware Workstation

1. Go to this link: [Download VMware Workstation](#)
2. Click on **VMware Workstation Pro 17 for Windows**
3. Click on the highest version available. (17.6.3 as of last update)
4. On the next screen, click on the blue link inside *I agree to the Terms and Conditions* to open a new tab with their terms and conditions. Do not close this window.
5. Back in the previous window/tab that's still open, now click the checkbox for *I agree to the Terms and Conditions*.
6. Click on the little cloud icon next to the name of the download file.
7. Click the **Yes** button when the dialog box pops up.
8. In the *Trade Compliance and Download Conditions* screen, fill in the following school information:
 - i. Address1: **1750 Finch Ave E**
 - ii. City: **North York**
 - iii. State/Province: **Ontario**
 - iv. Country: **Canada**
 - v. Zip/Postal Code: **M2J 2X5**
9. **Finally** click on the cloud icon again to begin the actual download.
10. If you ever need to download the software again, you won't have to go through any of the registration mess. Follow Part 3, Steps 1-3, then Part 9.

Part 4: Install VMware Workstation

1. Find the downloaded .exe file and run the installer.
2. Follow all prompts and install.

Part 5: Launch VMware Workstation

1. Open **VMware Workstation** from your desktop or the Start Menu.
2. When asked, paste in your serial key.
3. Proceed to the next Investigation.

Investigation 3: VM1 Installation - Windows Server 2025 Datacenter (*srv1*)

- Hypervisor: **VMware Workstation**
- Name: **srv1-cjohnson30**
- RAM: **16 GB**
- CPU: **6 cores**
- Storage: **250 GB**
- Networking: **2 NICs**
- ISO: **Windows Server 2025**

Flowchart Visualization of Investigation 3



Part 1: Setup Instructions

1. In the main window, you should see a large + symbol icon titled **Create a New Virtual Machine**. Click it.
2. In the new dialog box, keep *Typical* selected and click the **Next** button.
3. On the next screen, *Guest Operating System Installation*, do the following:
 - i. Select *Installer disc image file (ISO)*:
 - ii. Now click **Browse**.
 - iii. Navigate to where you saved your **Windows Server 2025 Datacenter** downloaded ISO and select it.
 - iv. Once selected, the previous screen should now say:

Windows Server 2025 detected.

This operating system will use Easy Install.
 - v. If it doesn't, you haven't selected the right file, or your download was corrupted. **Ask for help**.
 - vi. Click **Next**.
4. On the "Easy Install Information" screen, do the following:
 - i. Paste in your serial key.
 - ii. Version of Windows to install: Select *Windows Server 2025 Datacenter*
 - iii. Personalize Windows:
 - a. Full Name: **Administrator**

- b. Password (both fields): Select a strong password **you will remember**. You will use this same password for all VMs in this course.
- c. **Do not select "Log on automatically"**.
- d. Click **Next**.

5. On the "Name the Virtual Machine" screen, do the following:

- i. Virtual machine name: **srv1-senecausername**

Explanation: For example, if my Seneca e-mail address is `cjohnson30@myseneca.ca`, then my Seneca username is `cjohnson30`. This would give me a VM name of `srv1-cjohnson30`.

- ii. Location: If using an external SSD (like with our lab computers), click **Browse** and navigate to your external SSD.
 - a. Create the following directory structure in your SSD: `OSM620 > Virtual Machines/srv1-cjohnson30`
 - b. Select this new `srv1-cjohnson30` folder.
 - c. Make sure you now see this change in the Location field.

Example: `Z:\OSM620\Virtual Machines\srv1-cjohnson30`



- iii. Click **Next**.
6. On the "Specify Disk Capacity" screen, do the following:
 - i. Maximum disk size (GB): **250**
 - ii. Select *Split virtual disk into multiple files*.
 - iii. Click Next.
 7. On the "Ready to Create Virtual Machine" screen, do the following:
 - i. Click on **Customize hardware...**

8. On the new "Hardware" screen, do the following:
 - i. Select *Memory*, and change the value to: **16384**
 - ii. Select *Processors*, and change:
 - a. Number of processors: **1**
 - b. Number of cores per processor: **6**
 - c. Virtualize Intel VT-x/EPT or AMD-V/RVI: **Checked**
 - d. Virtualize CPU performance counters: **Unchecked**
 - e. Virtualize IOMMU (IO memory management unit): **Checked**
 - iii. Select *Network Adapter* and confirm:
 - a. *Connected at power on*: **Checked**
 - b. *NAT*: **Checked**
 - iv. Click on the **Add...** button on the bottom left of the *Hardware* window.
 - a. Select *Network Adapter* and click **Finish**.
 - b. Back in the *Hardware* window, click on *Network Adapter 2*.
 - c. Under *Network connection*, click the **Custom: Specific virtual network** radio button.
 - d. Just below that, click the drop-down (it likely says *VMnet0* by default). Find and select **VMnet10**.
 - e. Click **Close**.
9. Back in the "Ready to Create Virtual Machine" screen, click **Finish**.
10. The virtual machine should launch.
11. If you get a dialog box about *Side channel mitigations*, check the box for *Do not show this hint again* and click **OK**.
12. Your new Virtual Machine should now finish creating and then turn on and begin the OS installation.

13. Windows installation is automated at this point and won't require any input from you. It may restart several times.

Time Note: Installation may take some time.

Feel free to get some caffeine or make a sandwich.

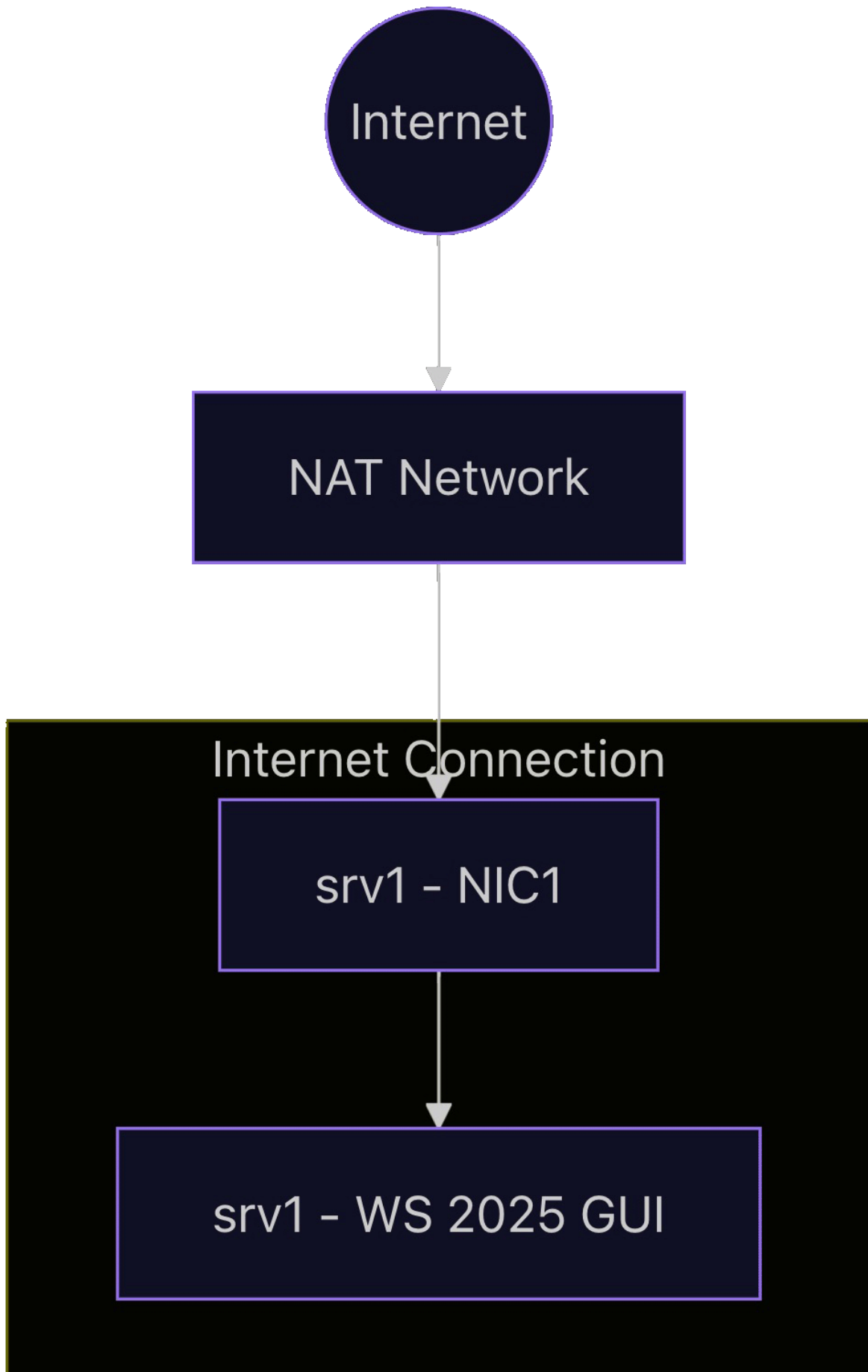
14. Eventually, you will be presented with the desktop and the VMware Tools installer having completed and asking if you'd like to restart. Choose **Yes**.
15. Once you've restarted, your installation is complete.

Investigation 4: Post-Installation Tasks (*srv1*)

After installing a new operating system, there are always a number of **post-installation tasks** to complete. **These aren't optional!**

Overview Check-In: *srv1* Internet Connection

Let's take a quick look at an overview of our NAT Internet connection as it currently stands. It's important to understand how everything is connected.



Part 1: Applying Time Zone Settings

This one is fairly straight-forward. Having the proper time zone set (EST) is essential for proper time keeping and ensuring encrypted webpages connect properly.

1. In the *Server Manager* application, click on **Local Server** in the left-hand menubar.
2. In the main *Properties* area, on the right-hand column, look for the *Time Zone* line. It should say **(UTC-05:00) Eastern Time (US & Canada)**.
3. If the *Time Zone* line item doesn't say the above, click on the displayed time zone and change it to UTC-05:00 as seen above.

Part 2: Server Name Change

The default name applied to your new server will be semi-randomized. For proper identification (and to not wonder which server you're on when you have several), we're going to change this.

1. In the *Server Manager* application, click on **Local Server** in the left-hand menubar.
2. In the main *Properties* area, on the left-hand column, look for the *Computer name* line.
3. Click the current computer name.
4. In the *System Properties* dialog box that pops up, find the **Change** button and click it. (Ignore the *Computer Description* field. It's tempting, but wrong!)
5. In the new *Computer Name/Domain Changes* dialog box that pops up, find the *Computer name* field. Replace it with **srv1-SenecaUsername**.
6. When you click **OK**, the system will warn you about restarting. Choose to

restart the system when asked.

7. Once you've restarted and logged back in, go back to the *Server Manager* from Part 1 and double-check your new computer name is correct. **Do not skip this step!**
8. If it is, you're done!

Part 3: Windows Activation

Activating Windows unlocks certain settings and features. Since you've used your valid serial key (right?), you can activate with Microsoft easily.

1. In the *Server Manager* application, click on **Local Server** in the left-hand menubar.
2. In the main *Properties* area, on the right-hand column, look for the *Product ID* line.
3. Click on the **Not activated** link.
4. Follow the instructions in the popup dialog box. If unable to activate easily, **ask your professor for help.**

Part 4: Installing OS Updates

A critical part of a security-conscious mindset is running regular updates. **This is NOT something you do only once at the start of installation.** You should be running these regularly to keep up to date with security fixes and zero-day exploits.

1. If you're already in the *Settings* application, look through the left menubar for **Windows Update** and click it. (Otherwise, click on the *Start* menu and search for **Updates**.)
2. In the *Windows Update* main screen, scroll down to **Advanced Options** and click it.

3. The very first option is *Receive updates to other Microsoft products*. Toggle this from **Off** to **On**.
4. At the top of the screen, where it says *Windows Update > Advanced Options*, click **Windows Update** to go back to the previous screen.
5. You will likely already see updates ready. Click on **Download & install all**.
6. As you might expect, this can take a while. Timing will depend on your Internet connection, how fast your computer is, how fast your SSD is, and how many updates there are. Please be patient. Your computer may restart.
7. Once updates have begun, take a break while it does its thing. Grab a drink, make a sandwich, text a friend.
8. After updates are complete, go back into *Windows Update* and click **Check for updates** again. There may be (and often times are) more.
9. If there are more updates, complete Steps 5-8 again until there are no more updates available.
10. In *Windows Update*, scroll back down to **Advanced options** again and click it.
11. Inside *Windows Update > Advanced options* scroll down to **Optional updates** and click it.
12. Select all available updates that appear (you may have to expand some lists).
13. Click **Download & install**.
14. Once all updates are complete, restart your VM if asked, then move onto the next section.

Part 5: Internet Connectivity Check w/Edge

We'll double-check we can access the Internet using the built-in Microsoft Edge application.

1. Open *Microsoft Edge*.
2. Go through the first-run questions.
3. When able, use the browser to navigate to **eff.org**.
4. If the website loads, move on to the next step. If not, **ask your professor for help**.

Part 6: Download and Install Firefox

There are a ton of feature and privacy reasons *not* to use Microsoft Edge. Instead, we'll download and install **Mozilla Firefox** and use that going forward.

1. Open *Microsoft Edge*.
2. Navigate to: **Download Mozilla Firefox**
3. Wait for the installer to finish, then open it.
4. Follow the installer instructions.
5. Once complete, open *Firefox* and navigate to **eff.org** to check everything is okay.
6. Close *Microsoft Edge* forever.

Investigation 5: VM2 Installation - Windows Server 2025 Core (*srv2*)

- Hypervisor: **VMware Workstation**
- Name: **srv2-cjohnson30**
- RAM: **4 GB**
- CPU: **2 cores**
- Storage: **250 GB**

- Networking: **2 NICs**
- ISO: **Windows Server 2025**

Flowchart Visualization of Investigation 5



Part 1: Setup Instructions

1. In the main window, you should see a large + symbol icon titled **Create a New Virtual Machine**. Click it.
2. In the new dialog box, keep *Typical* selected and click the **Next** button.
3. On the next screen, *Guest Operating System Installation*, do the following:
 - i. Select *Installer disc image file (ISO)*:
 - ii. Now click **Browse**.
 - a. Navigate to where you saved your **Windows Server 2025 Datacenter** downloaded ISO and select it.
 - b. Once selected, the previous screen should now say:

Windows Server 2025 detected.

This operating system will use Easy Install.
 - c. If it doesn't, you haven't selected the right file, or your download was corrupted. **Ask for help**.
 - iii. Click **Next**.
4. On the "Easy Install Information" screen, do the following:
 - i. Paste in your serial key.
 - ii. Version of Windows to install: Select *Windows Server 2025 Datacenter (Core)*
 - iii. Personalize Windows:
 - a. Full Name: Administrator

- b. Password (both fields): Select a strong password **you will remember**. You will use this same password for all VMs in this course.
 - iv. **Do not select "Log on automatically"**.
 - v. Click **Next**.
- 5. On the "Name the Virtual Machine" screen, do the following:
 - i. Virtual machine name: **srv2-senecausername**

Explanation: For example, if my Seneca e-mail address is `cjohnson30@myseneca.ca`, then my Seneca username is `cjohnson30`. This would give me a VM name of `srv2-cjohnson30`.
 - ii. Location: If using an external SSD (like with our lab computers), click **Browse** and navigate to your external SSD.
 - a. Create the following directory structure in your SSD: *OSM620 > Virtual Machines > srv2-cjohnson30*
 - b. Select this new *srv2-cjohnson30* folder.
 - c. Make sure you now see this change in the Location field. (Example: *Z:/OSM620/Virtual Machines/srv2-cjohnson30*)
 - iii. Click **Next**.
- 6. On the "Specify Disk Capacity" screen, do the following:
 - i. Maximum disk size (GB): **250**
 - ii. Select *Split virtual disk into multiple files*.
 - iii. Click Next.
- 7. On the "Ready to Create Virtual Machine" screen, do the following:
 - i. Click on **Customize hardware...**

8. On the new "Hardware" screen, do the following:
 - i. Select *Memory*, and change the value to: **4096**
 - ii. Select *Processors*, and change:
 - a. Number of processors: **1**
 - b. Number of cores per processor: **2**
 - c. Virtualize Intel VT-x/EPT or AMD-V/RVI: **Checked**
 - d. Virtualize CPU performance counters: **Unchecked**
 - e. Virtualize IOMMU (IO memory management unit): **Checked**
 - iii. Select *Network Adapter* and confirm:
 - a. *Connected at power on*: **Checked**
 - b. *NAT*: **Checked**
 - iv. Click on the **Add...** button on the bottom left of the *Hardware* window.
 - a. Select *Network Adapter* and click **Finish**.
 - b. Back in the *Hardware* window, click on *Network Adapter 2*.
 - c. Under *Network connection*, click the **Custom: Specific virtual network** radio button.
 - d. Just below that, click the drop-down (it likely says *VMnet0* by default). Find and select **VMnet10**.
 - e. Click **Close**.
9. Back in the "Ready to Create Virtual Machine" screen, click **Finish**.
10. The virtual machine should launch.
11. If you get a dialog box about *Side channel mitigations*, check the box for *Do not show this hint again* and click **OK**.
12. Your new Virtual Machine should now finish creating and then turn on and begin the OS installation.

13. Windows installation is automated at this point and won't require any input from you. It may restart several times.

Time Note: Installation may take some time.

Feel free to get some caffeine or make a sandwich.

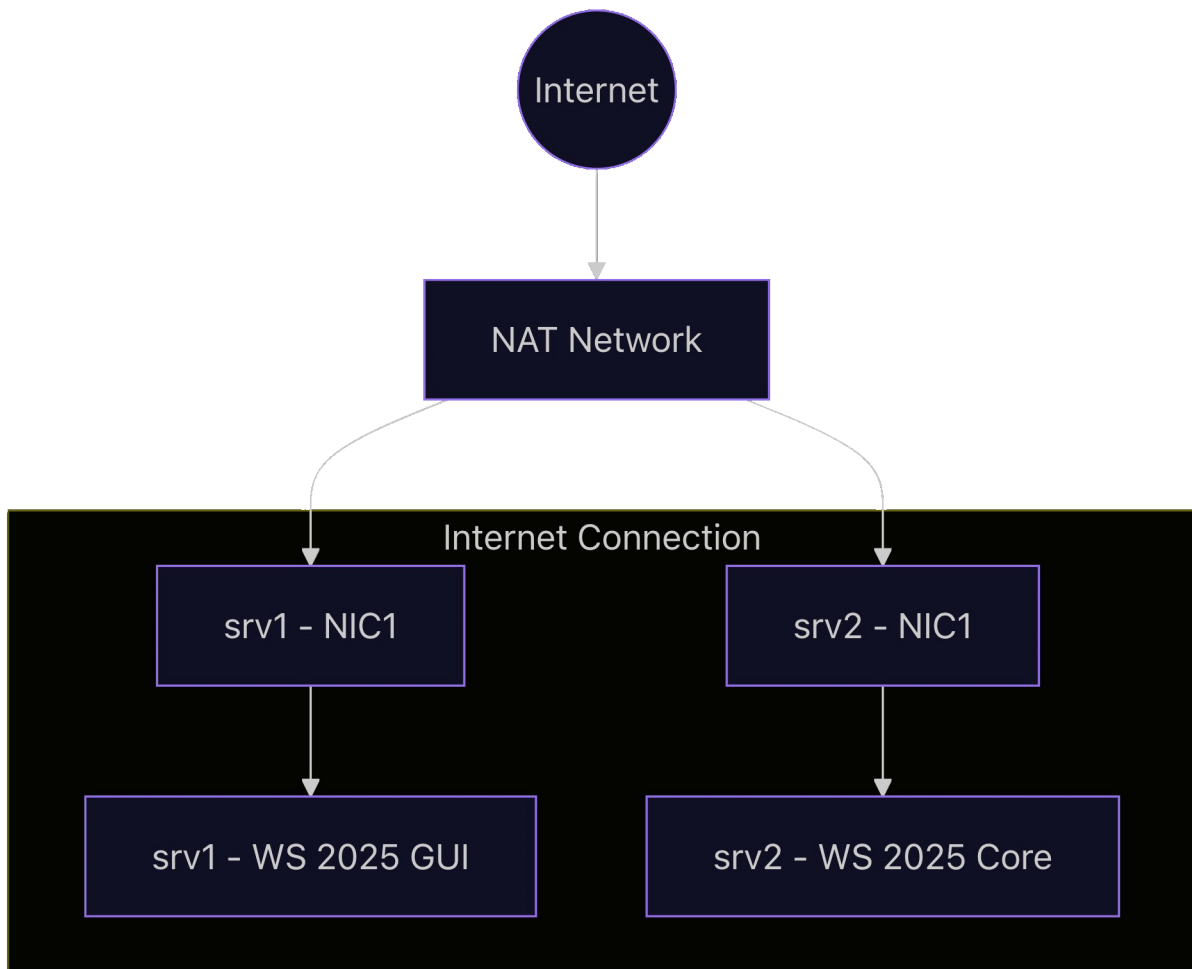
14. Eventually, you will be presented with the Command Prompt window open to the *SConfig* text-based application, and the VMware Tools installer having completed and asking if you'd like to restart. Choose **Yes**.
15. Once you've restarted, your installation is complete.

Investigation 6: Post-Installation Tasks (*srv2*)

After installing a new operating system, there are always a number of **post-installation tasks** to complete. **These aren't optional!**

Overview Check-In: *srv2* Internet Connection

Let's take a quick look at an overview of our NAT Internet connection as it currently stands since our newest server addition. It's important to understand how everything is connected as we continue to build out our VMs.



Part 1: Applying Time Zone Settings

This one is fairly straight-forward. Having the proper time zone set (EST) is essential for proper time keeping and ensuring encrypted webpages connect properly.

1. In the *SConfig* application, select Option 9 (*Date and Time*). Use your keyboard.
2. The *Date and Time* dialog box pops up.
3. Look for the *Time Zone* line. It should say **(UTC-05:00) Eastern Time (US & Canada)**.
4. If the *Time Zone* line item doesn't say the above, click on the *Change time zone...* button and change it to UTC-05:00 as seen above.

5. Click **OK** to close out of *Date and Time*.
6. Back in *SConfig*, choose Option 9 again to confirm your changes have stuck. If yes, continue to Part 2.

Part 2: Server Name Change

The default name applied to your new server will be semi-randomized. For proper identification (and to not wonder which server you're on when you have several), we're going to change this.

1. In the *SConfig* application, select Option 2 (*Computer name*). Use your keyboard.
2. In the new *Computer name* screen, enter your new computer name in the waiting text field: **srv2-SenecaUsername**
3. Press **Enter** on your keyboard to confirm the change.
4. The system now asks you about restarting. Enter **Y** to choose yes and hit the **Enter** key to confirm.
5. Once you've restarted and logged back in, go back to the *Computer name* screen from Part 2 and double-check your new computer name is correct.
Do not skip this step!
6. If it is, you're done!

Part 3: Windows Activation

Activating Windows unlocks certain settings and features. Since you've used your valid serial key (right?), you can activate with Microsoft easily.

1. In the *SConfig* application, select Option 11 (*Windows activation*). Use your keyboard.
2. In the new *Windows activation* screen, enter **2** (*Activate Windows*) and hit **Enter** to begin the activation process with Microsoft.
3. Follow the instructions on screen. If unable to activate easily, **ask your professor for help**.

Part 4: Installing OS Updates

A critical part of a security-conscious mindset is running regular updates. **This is NOT something you do only once at the start of installation.** You should be running these regularly to keep up to date with security fixes and zero-day exploits.

IMPORTANT: If Windows Update fails with errors, you may have Internet connectivity issues. Ask your professor for help!

1. In the *SConfig* application, select Option 5 (*Update settings*). Use your keyboard.
2. In the new *Update setting* screen, enter **5** (*Opt-in to Microsoft Update*) and hit **Enter** to confirm.
3. The next screen will ask you confirm again. Enter **y** and hit **Enter** to continue. (Then, hit **Enter** again.)
4. In the *SConfig* application, select Option 6 (*Install updates*). Use your keyboard.
5. In the new *Install updates* screen, enter **1** (*All quality updates*) and hit **Enter** to confirm.
6. After a short check, you are asked which updates to install. Select **a** to install all updates and hit **Enter** to confirm.
7. As you might expect, this can take a while. Timing will depend on your Internet connection, how fast your computer is, how fast your SSD is, and how many updates there are. Please be patient.
8. Once updates have begun, take a break while it does its thing. Grab a drink, make a sandwich, text a friend.
9. If asked to restart, choose **yes**.
10. After updates are complete, go through Steps 4-6 again. Do so until the system tells you there are no new updates. (It may take a few cycles to get them all.)
11. When complete, shut down *srv2* safely. Use the on-screen menu options in

SConfig to do so.

That's it! Now you're done. Congratulations!

Lab 1 Sign-Off

It is extremely important that you complete Lab 1 correctly as these Windows Server installs will be the platform on which the rest of the course will be completed.

When you have completed Lab 1, ask your instructor to come and check your installation. This must be done in class. They will ask you to complete a set of tasks/commands. If everything has been completed correctly, your instructor will mark your Lab 1 as complete.

Sign-Off Checklist

1. Open VMware Hardware Settings window for *srv1* and *srv2*.
2. *srv1* is currently running and logged in.
3. *srv2* is currently running and logged in.
4. No new updates available for both VMs.
5. Server names have been applied correctly.
6. Time zone is in EST.
7. Servers have been properly activated.
8. *srv1*: Firefox installed and open.

Lab 2 - Hyper-V, NAT, and Windows Clients

Lab Preparations

Purpose of Lab 2

In this lab, you'll move beyond basic VM deployment and learn to create nested virtualization environments. This is a crucial skill for modern sysadmins and cloud architects. You'll enable the Hyper-V role on your Windows Server, build a secure internal network using Hyper-V switches, and deploy Windows 11 client VMs within your existing server VM. You'll also set up NAT and routing, allowing your nested clients to access the internet safely through your server.

By the end, you'll have a working mini-enterprise: one physical computer running a server VM, which itself is hosting client VMs. Just like real-world cloud and enterprise setups! You'll gain hands-on experience with Hyper-V, RRAS, and NAT, and build a technical foundation for future labs in DHCP, DNS, and security.

Objectives

By the end of this lab, you will be able to:

- Enable the Hyper-V role on a Windows Server 2025 VM and configure virtual networking.
- Build and configure a Hyper-V External Virtual Switch bound to a specific NIC.
- Set up Routing and Remote Access (RRAS) for network address translation

(NAT) between internal and external networks.

- Deploy Windows 11 client VMs inside Hyper-V, configuring hardware and storage as needed.
- Assign static IPs and ensure that internal client VMs can reach the internet via NAT on your server.
- Practice real-world sysadmin tasks: managing nested VMs, routing, and internal networks.

Minimum Requirements

Before beginning, you must have:

1. Completed Lab 1 in full, with both srv1 and srv2 working, activated, and updated.
2. Attended the Week 2 lecture, or carefully read through the Week 2 slides.
3. Downloaded the Windows 11 Education ISO and your product key (from Lab 1).
4. Access to VMware Workstation and your external SSD with at least 200 GB of free space remaining.
5. Your printed OSM620 Lab Logbook for notetaking and command recording.
6. A basic understanding of server roles, virtual networking, and Windows installation processes.
7. Caffeine or snack of your choice: nested virtualization can be a long process!

Investigation 1: Hyper-V Role Installation and Switch Setup (*srv1*)

In this investigation, you will install the Hyper-V role on *srv1* and create a Hyper-V virtual switch that uses your existing Internal Network adapter.

Your Hyper-V switch will be named Internal Network (Hyper-V).

Part 1: Install the Hyper-V Role

1. Open the **Server Manager** application and go to **Add Roles and Features**.
2. **Installation Type:** Keep Role-based or feature-based installation selected.
3. **Server Selection:** Ensure your local server (*srv1*) is selected.
4. **Server Roles:** Check **Hyper-V**.
 - i. When prompted to "Add features that are required for Hyper-V", click **Add Features**.
5. **Features:** Leave defaults (the *Hyper-V Management Tools* are already included).
6. **Hyper-V page:**
 - i. When asked about Virtual Switches, select **Internal Network** only.
 - ii. Migration for live migrations: leave unchecked.
 - iii. Default Stores: leave defaults.
7. Confirmation: Check "Restart the destination server automatically if required".
8. Click **Install**.
9. Wait for installation to complete. If prompted, allow the reboot. Log back in

when finished.

Note: It's normal for *Server Manager* to take a minute or two after reboot to show the Hyper-V tile.

Part 2: Rename the Hyper-V Virtual Switch

As part of our Hyper-V role installation, we selected the *Internal Network* (Part 1, Step 6.1).

This has created a virtual switch on the server. For easy management, we'll rename it.

Our new virtual switch name will be: `HQ Network`

1. In Server Manager, click Tools (top-right) and select Hyper-V Manager.
2. In the left pane, click your server name (e.g., srv1-...).
3. In the right Actions pane, click Virtual Switch Manager....
4. Under the Virtual Switches column on the left, look for the switch that's already been created. It will likely have a name similar to: `Intel(R) 82574L Gigabit Network Connection - Virtual Switch`
5. In the Virtual Switch Properties window:
 - i. Name: Change to `HQ Network`
 - ii. Ensure the **External network** radio button is selected.

Reminder: This is the adapter you renamed earlier to Internal Network and set to 10.0.`UID`.1.

- iii. Ensure "*Allow management operating system to share this network adapter*" is checked.
- iv. Click OK. If you see a warning about temporary network disruption, click Yes.

Naming note: You are creating an External Hyper-V switch, but you're binding it to the Windows NIC you named Internal Network.

That's correct.

You will end up with a third network adapter called: vEthernet (HQ Network)

6. Go back to *Server Manager > Local Server* and refresh. You should now see the following two networks:
 - **External Network:** Assigned by DHCP
 - **vEthernet (HQ Network):** IPv4 address assigned by DHCP, IPv6 enabled

Part 3: Configure Network Setting for Hyper-V Switch

Now that we've created the Hyper-V network adapter, we have to give it an IP address. This will let our Hyper-V VMs talk to the server, and eventually, bridge our Internet connection.

1. In *Server Manager > Local Server*, click on the entry next to **vEthernet (HQ Network)**. This opens the *Network Connections* window.
2. Verify you now see a third network adapter: vEthernet (HQ Network)
3. Right-click it → Properties and do the following:

- i. Internet Protocol Version 6 (TCP/IPv6): **Unchecked**
 - ii. Internet Protocol Version 4 (TCP/IPv4): Select and click **Properties**
 - a. IPv4 Address: **10.0. UID .1**
 - b. Subnet Mask: **255.255.255.0**
 - c. Leave all other fields blank and click OK.
4. Back in *Server Manager* > *Local Server*, refresh. You should now see the following two networks:
- **External Network**: Assigned by DHCP
 - **vEthernet (HQ Network)**: 10.0. UID .1

Investigation 2: RRAS and NAT

Our server now has access to two networks:

- **External Network**: Connection to the Internet
- **vEthernet (HQ Network)**: Internal network only between *srv1* and the Hyper-V VMs we'll create in Investigation 3. This network *does not* have access to the Internet.

If we want to give our Hyper-V machines Internet access, we need to route network traffic between the two networks using *srv1* as a bridge. This keeps our Hyper-V machines on a secure network, and Internet access is handled and monitored by our Windows Server.

This will involve setting up routing and network address translation (NAT).

Welcome to Microsoft's **Routing and Remote Server (RRAS)**.

Part 1: Installing the Remote Access Role

1. Open the **Server Manager** application and go to **Add Roles and**

Features.

2. **Installation Type:** Keep Role-based or feature-based installation selected.
3. **Server Selection:** Ensure your local server (*srv1*) is selected.
4. **Server Roles:** Check **Remote Access**.
5. **Features:** Leave defaults (the *Hyper-V Management Tools* are already included).
6. **Remote Access page:** Click *Next*.
 - i. The **Add Roles and Features Wizard** dialog box appears. Leave defaults and select *Add Features*.
7. **Remote Access > Role Services page:** Check the following and click *Next* when ready:
 - Direct Access and VPN (RAS)
 - Routing
8. Keep defaults and select *Next* on the following pages:
 - i. Web Server Role (IIS)
 - ii. Web Server Role (IIS) > Role Services
9. Confirmation: Click **Install**.
10. In the **Results** page: When installation has completed, click on the "*Open the Getting Started Wizard*" link.
11. The wizard opens. Make no changes and close it. (This just removes the system nagging.)
12. Back in **Results**, click *Close*.

Part 2: Enabling NAT with RRAS

You will now install the tools for routing and NAT bridging discussed at the top of this investigation using the **Routing and Remote Access Server** (RRAS) we've just installed.

1. Open the **Server Manager** application and go to **Tools > Routing and**

Remote Access.

2. This opens the *Routing and Remote Access* application.
3. In the right-hand column, find SRV1, right-click it, and select **Configure and Enable Routing and Remote Access**.
4. The setup wizard appears. Click *Next*.
5. **Configuration:** Select the *Custom configuration* radio button.
6. **Custom Configuration:** Select (checkbox) only the following:
 - NAT
 - LAN routing
7. **Summary window:** Click *Finish*.
8. The **Routing and Remote Access** service status dialog box appears. Click *Start service*.
9. It may take a few moments for the service to fully start. Be patient.
10. Once the *Routing and Remote Access* application reappears, move to the next part.

Part 3: Configuring NAT with RRAS

It's now time to configure the routing and NAT bridging between the two networks.

1. Open the **Routing and Remote Access** application.
2. In the right-hand column, expand *SRV1*, then expand *IPv4*.
3. Right-click on *NAT* and select *New Interface*.
4. Select the **External Network** entry and click *OK*.
5. In the *Network Address Translation Properties* window, select the following and click *OK* when ready:
 - Public interface connected to the Internet
 - Enable NAT on this interface

6. Right-click on *NAT* and select *New Interface* again.
7. Select the **vEthernet (HQ Network)** entry and click OK.
8. In the *Network Address Translation Properties* window, select the following and click OK when ready:
 - Private interface connected to private network
9. Back in the *NAT* window, you should see two created interfaces:
 - External Network
 - vEthernet (HQ Network)
10. Keep the application open on *NAT* and minimize. We'll come back to it later.
11. That's it! Move on to the next Investigation.

Investigation 3: Installing Windows Client 1 with Hyper-V (*client1*)

In this investigation, you'll create your first Hyper-V based virtual machines.

These will be Windows 11 client machines. Think of these as typical workstations in a corporate office.

Part 1: Creating the Hyper-V Virtual Machine

- Hypervisor: **Hyper-V**
- Name: **client1-SenecaUsername**
- RAM: **4 GB**
- CPU: **2 processors**
- Storage: **64 GB**
- NIC: **1** (default)
- Security: **Enable Trusted Platform Module**

- ISO: **Windows 11 Education**

1. Drag and drop the *Microsoft Windows 11 ISO* file from Lab 1 onto the desktop of *srv1*. It should copy the file there. If it doesn't, ask your professor for help before continuing.
2. Open the **Server Manager** application and go to **Tools > Hyper-V Manager**.
3. This opens the *Hyper-V Manager* application.
4. In the right-hand column, click on **New > Virtual Machine...**
5. The *New Virtual Machine Wizard* appears.
6. On the *Before You Begin* page, check the **Do not show this page again** option and click **Next**.
7. **Specify Name and Location:**
 - Name: **client1-SenecaUsername**
 - Leave the rest at their defaults.
8. **Specify Generation:** Leave defaults.
9. **Assign Memory:** 4096 MB
10. **Configure Networking:**
 - Connection: **HQ Network**
11. **Connect Virtual Hard Disk:**
 - **Create a virtual hard disk** selected.
 - Size: **64 GB**
 - Leave the rest at their defaults.
12. **Installer Options:**
 - Select **Install an operating system from a bootable CD/DVD-ROM**
 - Select **Image file (.iso):**
 - Click **Browse** and select the Windows 11 ISO image file you copied to *srv1*.

13. **Summary:** Click *Finish*.

WARNING: Do not start the *client1* VM, yet! We need to modify it. See *Part 2*.

Part 2: Modifying *client1* VM for Windows 11

Despite Hyper-V being a Microsoft product and Windows 11 being a Microsoft product, Hyper-V doesn't automatically add everything needed to run Windows 11.

If you think this is silly... It is. Welcome to the wonderful world of IT.

1. In the **Hyper-V Manager** application, select the *client1* VM.
2. In the right-hand column, look for and click on the **Settings...* option.
3. This opens up the settings window for the *client1* VM. Change the following:
4. **Processor:**
 - Number of virtual processors: **2**
 - Make sure to click **Apply!**
5. **Security:**
 - Encryption Support > Enable Trusted Platform Module: **Checked**
 - Make sure to click **Apply!**
6. Click **OK** to close the VM settings window.

Part 3: Hyper-V Client Management

Knowing how to power on and off your Hyper-V VM is important and includes a few cool features.

In the **Hyper-V Manager** application, select the **client1** VM. (Single click. Do

not double-click.)

On the right-hand column, towards the bottom, you have:

- **Connect:** This *connects* to the VM. It will open a virtual monitor to the machine. Just like a normal computer, if the computer (VM) is off, connecting to it will do nothing but give you a blank screen.
- **Settings:** You used this in *Part 2*. Modify VM settings. Don't change anything unless told to.
- **Start:** Power on the VM. If you are already connected to it, you will see it power up. If not connected, it still powers up. Click on **Connect** to open a screen to it.

When the VM is powered on, you have additional options in that column:

- **Turn Off:** This is the equivalent of pulling the power plug. Don't do this unless the VM is fully frozen and not responding from commands. (This is the "*Nuke it from orbit.*" option. It works, but it's overkill and may cause damage.)
- **Shut down:** This is the polite version of the option above. Hyper-V will send a gentle "Please shut down." command to the Windows 11 client. If the Windows 11 client is working, it'll close all applications and shut down properly. Hyper-V detects the OS having shut down and then powers off the VM itself. (**This is your preferred option.**)
- **Reset:** Equivalent to pressing the *Reset* physical button on a computer. Yanks the power and then turns it back on. AKA "*Nuke and reload.*" (Again, don't use unless unresponsive.)
- The **Start** button disappears when the VM is on, which makes sense.

There are other options, but those are enough to get by for now.

IMPORTANT: Always safely shut down all Hyper-V VMs before shutting down your Windows Server!

Part 4: Installing Windows 11 (*client1*)

1. Once the Windows 11 Setup screen appears:
2. On *Select language settings*, keep the defaults and click **Next**.
3. On *Select keyboard settings*, keep the defaults and click **Next**.
4. On *Select setup option*, do the following:
 - i. Select *Install Windows 11*
 - ii. Check the box next to *I agree everything will be deleted, including files, apps, and settings*
 - iii. Click **Next**.
5. On the *Product key* page, enter your product key and click **Next**.
6. On *Applicable notices and license terms*, click **Accept**.
7. On *Select location to install Windows 11*, keep the defaults and click **Next**.
8. On *Ready to install*, click **Install**.
9. The Windows Installer will now install the OS. This may take some time, and the percentage may freeze at certain points. Be patient.
10. When the installer finishes, Windows 11 will start up and you will be launched into the First-Run Setup. Go to Part 3.

Part 5: First-Run Setup

1. On *Is this the right country or region?*, select **Canada** and click **Yes**.
2. On *Is this the right keyboard layout or input method?*, stick with the default and click **Yes**.
3. On *Want to add a secondary keyboard layout?*, click **Skip**.

Note: If it has an Internet connection, Windows will now check for

available updates from the Internet. If it finds any, it will install them automatically.

If it doesn't have an Internet connection, this step will be skipped.

This process may take some time. Please be patient. Your VM may restart on its own.

Part 6: Account Creation

Now that language/keyboard and installer updates have been applied, it's time to create your account.

Note: Steps 1-2 may not be necessary. If you're asked for a name, start with Step 3.

1. On *Let's set things up for work or school*, select **Sign-in options**.
2. On this next screen, click on **Domain join instead**.
3. On *Who's going to use this device?*, enter your **Seneca username**, (Not your full name), then click **Next**.
4. On *Create a super memorable password*, enter the same password you've used for your other VMs and click **Next**.
5. Confirm it on the next screen and continue.
6. On *Now add security questions*, fill out three security questions, clicking **Next** after each.
7. On *Let Microsoft and apps use your location*, select **No**, then click **Accept**.
8. On *Find my device*, select **No**, then click **Accept**.
9. On *Send diagnostic data to Microsoft*, scroll down to select **Required only**, then click **Accept**.
10. On *Improve inking & typing*, select **No**, then click **Accept**.
11. On *Get tailored experiences with diagnostic data*, select **No**, then click

Accept.

Note: If it has a valid Internet connection, Windows will now check for *more* available updates. As before, if it finds any, it will install them automatically.

If not, it will simply continue.

This process may take some time. Please be patient. Your VM may restart on its own.

Once complete, you will be presented with a login screen. Move to the next part to continue.

Investigation 4: Post-Installation Tasks (*client1*)

In this investigation, we'll log in for the first time and run through several post-installation tasks both necessary and for user comfort.

Part 1: First Login

1. Enter your password to login. First-login may take a few minutes as your profile is set up.

Part 2: Network Configuration

1. Click on the *Start* button and search for **Network Connections**. Open it when found.
2. Find the only Ethernet adapter (it may have different names).
3. Right-click it → Properties and do the following:

- i. Internet Protocol Version 6 (TCP/IPv6): **Unchecked**
 - ii. Internet Protocol Version 4 (TCP/IPv4): Select and click **Properties**
 - a. IPv4 address: **10.0.UID.11**
 - b. Subnet mask: **255.255.255.0**
 - c. Default gateway: 10.0.UID.1
 - d. Preferred DNS server: 10.0.UID.1
 - e. Alternate DNS server: 8.8.8.8
 - f. Leave all other fields blank and click OK.
4. Open *Microsoft Edge* and navigate to **eff.org**. If the page loads, you're done!

Check it out: You are now using RRAS+NAT on your Windows Server to make this Internet connection happen.

In *srv1*, go back to the **Routing and Remote Access** application. Inside, go to *SRV1 > IPv4 > NAT*. The **External Network** entry now has non-zero numbers under the 'packets translated' columns. This is your routing and NAT from *client1* at work! Refresh the page to see the count go up.

Part 2: Setting the Time Zone

1. Login again.
2. Find the time on the bottom right of the screen (in the VM, not your host machine!)
3. Right-click on the time and select **Adjust date and time**.
4. Check the displayed time and time zone.
5. The time should match current time, and the time zone should say "(UTC-5:00) Eastern Time (US & Canada)".
6. If the time zone is wrong, change it to the value above.
7. Confirm your time zone and time matches local time.

Part 3: Setting Internal Computer Name

When we created our VM, we gave it the name *client1-SenecaUsername*. This only applies to how VMware Workstation sees the VM, not to how the internal Windows OS sees itself. We need to change that to match.

1. In the *Settings* window you still have open (or reopen it from the Search bar), look to the left-hand menu bar and select **System**.
2. The very first thing you see at the top is the current computer name. This is what we're going to change.
3. Click on the **Rename** text link.
4. In the *Name your device* field, enter your VM's name: **client1-SenecaUsername** (replacing *SenecaUsername* with your actual username)
5. Click **Next**.
6. You will now be asked if you'd like to restart. Click **Restart now**.
7. After restarting and logging back in, go into *Settings* > *System* and confirm your computer name is now **client1-SenecaUsername**.

Part 4: Windows Updates

A critical part of a security-conscious mindset is running regular updates. **This is NOT something you do only once at the start of installation.** You should be running these regularly to keep up to date with security fixes and zero-day exploits.

1. If you're already in the *Settings* application, look through the left menubar for **Windows Update** and click it.
2. In the *Windows Update* main screen, scroll down to **Advanced Options** and click it.

3. The very first option is *Receive updates to other Microsoft products*. Toggle this from **Off** to **On**.
4. At the top of the screen, where it says *Windows Update > Advanced Options*, click **Windows Update** to go back to the previous screen.
5. You will likely already see updates ready. Click on **Download & install all**.
6. As you might expect, this can take a while. Timing will depend on your Internet connection, how fast your computer is, how fast your SSD is, and how many updates there are. Please be patient. Your computer may restart.
7. Once updates have begun, take a break while it does its thing. Grab a drink, make a sandwich, text a friend.
8. After updates are complete, go back into *Windows Update* and click **Check for updates** again. There may be (and often times are) more.
9. If there are more updates, complete Steps 5-8 again until there are no more updates available.
10. In *Windows Update*, scroll back down to **Advanced options** again and click it.
11. Inside *Windows Update > Advanced options* scroll down to **Optional updates** and click it.
12. Select all available updates that appear (you may have to expand some lists).
13. Click **Download & install**.

Investigation 5: Installing Windows Client 2 with Hyper-V (*client2*)

In this investigation, you'll create your second Hyper-V based virtual machine. This will be another Windows 11 client.

To conserve resources and speed up installation, it's helpful to shutdown *client1* while doing this.

Part 1: Creating the Hyper-V Virtual Machine

Create a second Hyper-V virtual machine with the following settings:

- Hypervisor: **Hyper-V**
- Name: **client2-**`SenecaUsername`
- RAM: **4 GB**
- CPU: **2 processors**
- Storage: **64 GB**
- NIC: **1** (default)
- Security: **Enable Trusted Platform Module**
- ISO: **Windows 11 Education**

Part 2: Installing Windows 11 (*client2*)

Set this up exactly as you did with *client1*, with **two important changes**:

1. Computer Name: **client2-**`SenecaUsername`
2. IP address: **10.0.**`UID`**.12**

Part 3: Post-Installation Tasks (*client2*)

Don't forget to run your **Post-Installation Tasks**! These are the same as for *client1*

Once *client2* is installed and has a working Internet connection, you're done!

Lab 2 Sign-Off

It's essential to complete Lab 2 correctly, as the nested virtualization setup will be the core environment for all advanced labs. Everything you do from this point (DHCP, DNS, security, and domain controller work) will depend on this structure being solid and consistent.

When you finish Lab 2, ask your instructor for a sign-off. Your instructor will check your configuration, verify networking, and confirm all VMs are working as required.

Sign-Off Checklist

1. Hyper-V role is installed and running on srv1.
2. HQ Network (External Virtual Switch) exists and is bound to the correct NIC.
3. RRAS (Routing and Remote Access) is installed and NAT is configured for the two networks:
 - i. External Network: **Internet access (DHCP-assigned)**
 - ii. vEthernet (HQ Network): **10.0.UID.1/24**
4. client1 and client2 are both running inside Hyper-V, with the following:
 - i. Static IPs set: **10.0.UID.11** and **10.0.UID.12**, respectively.
 - ii. Default gateway set to **10.0.UID.1**
 - iii. Both can access the internet (test by browsing to eff.org).
 - iv. Both have correct computer names set.
 - v. Both have time zone set to EST.
 - vi. Both are fully updated via Windows Update.
5. Graceful shutdown: You can demonstrate safe start, shutdown, and reset procedures for Hyper-V VMs.
6. All relevant steps, commands, and decisions are recorded in your Lab Logbook.

Lab 3 - Security and Remote Connectivity

Lab Preparation

Purpose / Objectives of Lab 3

In this lab, you will conduct several Windows system administration tasks to secure your servers against would-be attackers and gain preliminary experience with the command line interface.

If you encounter technical issues, please contact your professor via e-mail or in your section's Microsoft Teams group.

Minimum Requirements

Before beginning, you must have:

1. Successfully completed Lab 2.
2. Attended the Week 3 lecture
3. Read through the Week 3 slides, and have them handy as a reference for concepts.
4. Your external SSD (or personal computer) with your VMs from Labs 1 and 2.
5. Your VM login credentials.
6. Optional, but recommended: Caffeine delivery system.

Key Concepts

Security: From the Beginning

In the not-too-distant past, companies would focus on getting their product and systems working and relegating security as their last step, often as an afterthought. When security is only considered at the end of a project, it's very difficult to remember all the ways in which your product interacts and things can get missed.

This created several high-profile breaches in the 90s and early 2000s, and our approach to security had to be reconsidered.

As a result, we now consider security **from the beginning**. As you create applications, add users to databases, create links between services, you *must* keep security in mind at every step of development. Securing as you go is the best method, but even something as simple as simply documenting unsecured parts of your code as you go can be enough (assuming you go back and fix them!)

Generally, we apply the concept of **Principle of Least Privilege** to security. Essentially, this boils down to locking everything down as much as possible and only allowing what and who you need through. Open access makes you a target. You'll be applying this principle to the firewall later in this lab.

We also take a look at defaults. Most systems and software come with pre-configured defaults to make out-of-the-box setup easy. This can take the form of a default username and password, default ports, etc. In a well secured system, these are often changed to avoid hack attempts. If you know the default, there's a high chance that hackers know it as well. You'll be changing some defaults in this lab.

This is not an exhaustive list of applied security, but it does give us a bit of working knowledge. You'll need it for this lab as well as in our later work.

Firewalls

In short, **a firewall is a utility that sits on your computer between your network connection and the rest of your system.**

Any application, service, or other data that is sent or received by your computer goes through your firewall first. The dominant network protocol is TCP/IP, which means we're dealing with *packets*.

A firewall looks at these packets.

To be clear, the firewall doesn't look *inside* packets, but just at the outside data like IP address and/or port destination, etc. The actual transmitted data is still secure and unread.

Generally with firewalls, we apply the *Principle of Least Privilege* by dropping all new connections by default, and allowing a few exceptions. This is known as **whitelisting**.

Editing Text Files

As you will sometimes be working in the Windows PowerShell command line environment, it is useful to learn at least one common method of editing text files.

Although programmers and developers usually use graphical IDE's to code and compile programs (Visual Studio, Sublime, Eclipse, etc), they can create source code using a text editor and compile their code directly on the server to generate executable programs (without having to transfer them for compilation or execution).

Developers very often use a text editor to modify configuration files. In this course, you will become familiar with the process of installing, configuring, and running network services. Text editors are an important tool to help this setup, but are also used to "tweak" or make periodic changes in service configurations.

The most readily-available command line text editor built into Windows is **Notepad**.

However, Notepad is not available in Server Core. To edit a text file in that environment, we have three main options:

1. Use PowerShell's object-oriented programming to send an edit directly into a text file. This is cumbersome and is not interactive (you don't see the text file on screen), but it is the only built-in option.
2. Transferring the file to a different computer that does have a text editor (like your Windows 11 client), modifying the file there, and transferring it back to your Server Core machine.
3. Using **Visual Studio Code** to connect remotely the environment. This is by far the coolest and most convenient option.

We'll be doing all three in this lab to show you how, but going forward, VS Code will be our go-to for interacting with Server Core in most instances. (You will be tested on all three options, so don't skip them!)

Investigation 1: Windows Defender (Firewall)

In this investigation, we're going to take a look at the Windows Defender firewall to see how firewall rules can be applied to secure our servers and let the few things we want to allow through.

It should be noted that the Windows Defender firewall *should not be your only*

defense. Production environments will often have managed networks that restrict access on a larger scale, some of which you will get into with your networking courses.

These work together, though we will only be focusing (mostly) on the Windows firewall in this course.

Part 1: Setting Up *srv2*'s NIC2 - Internal Network

In this part, we'll rename the second NIC and then give it a 10.x.x.x address so we can use it to communicate with our other servers and clients.

1. Login to *srv2*.
2. In the `sfconfig` application, select Option 8 (Network Settings).
3. You will have two network adapters. We're going to change their names for easier identification.
4. Find the adapter that has the external network IP address. It's usually the first one in the list. If you're not sure, ask.
5. Select that entry's number from the first column to go into it's options.
6. In *Network Adapter Settings*, select Option 4 (Rename network adapter).
7. Type the following and hit Enter: `External Network`
8. Repeat steps 3-7 for the second network adapter, and call it: `Internal Network`
9. Go back to the main `sfconfig` screen.
10. Select Option 15 to exit to the command line.

We're now going to apply network settings to our Internal Network adapter using PowerShell.

11. First, confirm the names of your adapters with the following command:

```
Get-NetAdapter
```

12. Now, assuming they look correct, run the following:

```
New-NetIPAddress -InterfaceAlias "Internal Network" -IPAddress  
10.0.0.2 -PrefixLength 24
```

13. Double-check your work by running:

```
Get-NetIPAddress -InterfaceAlias "Internal Network"
```

If it has the proper 10.x.x.x IP address, well done! Move on to **Part 2**.

If the network information is wrong, you can remove it and try again by running:

```
Remove-NetIPAddress -InterfaceAlias "Internal Network"  
-AddressFamily IPv4
```

Tip: To return to `sfconfig`, either run that command or type `exit`.

Part 2: Windows Server GUI (srv1) - Testing the IIS Role

On this server, we will take a look at a role that was installed in our last lab as part of RRAS, **IIS (a web server)**. This will be used only for testing connectivity between your other VMs and working with the firewalls.

This role is already running after Lab 2 and doesn't need any configuration. We'll be using it later to test network connections and firewall rules.

Let's test it locally (from within the same machine). We'll do so by pointing a

web browser at ourselves.

1. Inside *srv1*, open Firefox.
2. Navigate to: `http://127.0.0.1`

If you see the *Internet Information Services* splash page, then this role is working. If you don't, ask your professor for help.

i A note about servers, clients, Server Roles, and IIS:

The IIS Server Role installs a **web server** inside of *srv1*. A web server is one that serves many clients (web browsers). This one-to-many concept is known as the client-server model in IT spaces. Other VMs and computers can also access this test web page (though it is blocked by default, see the Investigations below).

Part 3: Testing VM Network Connectivity

As part of your Lab 1 environment setup, you tested your connection to the Internet from each of your 3 virtual machines.

It's now time to test if we can use each VM to connect to *each other*.

1. On your Windows Server GUI (*srv1*) VM, open Command Prompt.
2. Grab this VM's **Internal Network** IP address with the following command and write it down: `ipconfig`

Remember, your *Internal Network* is the non-Internet network you created with the **10.0.UID.1** address. We're running `ipconfig` to confirm it's still there.

3. On your **Windows Client (client 1) VM**, open Command Prompt.
4. Enter the following command to test our ability to talk to *srv1*: `ping`

"`srv1-ipaddress`" where *srv1-ipaddress* is the IP address from Step 2.

Example: `ping 10.0.40.1`

5. It doesn't work, does it? That's normal at this stage and we'll fix it below. For now, we'll try connecting to our web server on *srv1*. That should show us the two VMs can talk to each other.

6. Open *Firefox* on your **Windows 11 Client (client1)** and use the IP address from Step 2 (your *srv1* IP address) as the URL.

Example: `http://10.0.40.1`

7. Can you see the IIS splash page that you saw in *Investigation 1, Part 1, Step 2*? If so, you have connectivity!

8. Continue to **Part 4**.

Part 4: Windows Server GUI (srv1) - Applying Firewall Rules

We will now apply our security-conscious policy by configuring the firewall on this server.

At the moment, the server's firewall is configured using defaults. As mentioned above, **defaults are a security risk** as they are known to everyone and can be used against you. If an attacker *doesn't* know your configuration, it's harder for them to know what's open and what's not and how to attack.

As you saw from **Part 2**, ping between *client1* and *srv1* didn't work.

By default, the ability to ping a server is turned off. A ping (a type of ICMP packet) is typically used to see if you can get a response from a server. Having a server respond let's an attacker know there's a machine there that they can

then try to break into.

Imagine a burglar knocks on a random brick wall (sends a ping). If you (the server) knock back (send a ping reply), the burglar now knows there's a person there who probably has some nice stuff! Don't knock back, and the burglar has no idea what's on the other side of that wall and will move on.

We want to turn on ping so we can test connections between our machines, but we have to be careful. Turning that on through the firewall too broadly opens us up to that vulnerability.

We are going to turn it on *only* for our local network. Our VMs will be allowed to ping each other, but anything outside of our subnet (10.0.0.0/24) can't. Best of both worlds.

To do this, we're going to work with the **Windows Defender Firewall** on srv1.

1. In the *Server Manager* application, go to *Local Server*.
2. In the *Properties* section of this page, look for the **Microsoft Defender Firewall** line item.
3. Next to the line, it should say **Public: On**. Click on this.
4. The *Windows Security > Firewall & Network protection* application opens.
5. On this screen, you will see three networks: *Domain network*, *Private network*, and *Public network*.
6. We'll spend more time with these in later weeks, but for now, all three should say: **Firewall is on**.
7. Below this, click on the link that says: **Advanced settings**
8. This opens the *Windows Defender Firewall with Advanced Security* application.
9. On the first page, you'll notice the same three profiles: *Domain Profile*, *Private Profile*, and *Public Profile*.

10. All three have the same overall rules:
11. **Windows Defender Firewall is on:** This just confirms the firewall is active for this network profile.
12. **Inbound connections that do not match a rule are blocked:** This means that all incoming connections and requests (like trying to ping this machine) are blocked *by default*. If you want to allow a certain type of connection or service into your server, you have to make a specific rule for it. This is whitelisting and our *Principle of Least Privilege* in action.
13. **Outbound connections that do not match a rule are allowed:** This means all network data leaving the server is automatically allowed. The logic here is that if the server is the one deciding the send out information, it's likely fine. (Note: The only time we ever whitelist outbound connections is for specialized security settings like government compliance.)
14. At the moment, we're dealing with the *Public Profile* context. Let's allow ICMP ping!
15. On the left-hand menu bar, click **Inbound Rules**.
16. This loads a ton of already-defined rules. Thankfully, the one we need has already been defined. We just need to turn it on.
17. At the top of this area, click **Name** to order the list by name (this is not the default).
18. Now, scroll down until you can find the following and double-click it: **Core Networking Diagnostics - ICMP Echo Request (ICMPv4-In), Private, Public**.
19. **Note:** There are two of these rules! Look for the one that has **Private, Public** under the Profile header. The wrong one says Domain.
20. This opens this rule's configuration window, starting on the *General* tab.
21. In the *General* tab, look for the following and check the box next to it:
Enabled
22. Click **Apply**, then **OK**.
23. Back in the main *Inbound Rules* window, you should see that same rule line

now says **Yes** under the *Enabled* header.

24. You've enabled ICMP ping! Let's go test it.
25. Switch back to your client1 machine, and run the same ping you ran earlier pointing to *srv1*. (Refer to *Part 2, Step 4*.)
26. Does it work?
27. If it does, congratulations! You've just enabled ping for connectivity checking to *srv1* and gone through your first foray into the Windows Firewall.

i A note for later labs: By default, this rule is set to only allow incoming pings (ICMP requests) from computers on your **VM network**—that is, other virtual machines on the same VMware NAT or host-only network as your server—not from the wider Internet or any physical computers in the classroom.

You can see this by checking the **Scope** tab in the rule's properties, where "Remote IP address" is set to "Local subnet."

We'll spend more time on how **Scope** and **Private/Public profiles** affect your firewall rules in our *Secondary Network* assignment.

 **Lab Question:** Why did we *not* have to do this for our IIS web server setup?

In the **Inbound Rules** list, scroll through to see if you can find the rule that's allowing web server pages to be requested from *srv1*.

Write down the name of the rule in your Lab Logbook when you find it (it's not called IIS) and explain why you think you didn't have to enable this rule yourself. Think back to when you installed the IIS Server Role.

Part 5: Windows Server Core (srv2) - Applying Firewall Rules

Let's apply the same incoming ping firewall rule to our Server Core machine so we can check it's network connectivity as well.

1. Login to your Server Core (*srv2*) machine.
2. The *sfconfig* text-based application automatically launches.
3. Select option 8 to find this machine's **Internal Network** IP address. Write it down.
4. Back in your Windows 11 Client (*client1*), try to ping this address. Does it work?
5. Just as in *srv1*, it doesn't.
6. Go back to *srv2*.
7. If you're still in the *Network settings* page, leave the field blank and hit **Enter** to go back to the main screen.
8. Select option 15 to exit to PowerShell.
9. As there's no GUI, we need to use PowerShell for firewall management.
10. Let's take a look at the existing rules, just like in Part 2. Run the following PowerShell command:

```
Get-NetFirewallRule | Where-Object DisplayName -Like '*ICMPv4-In*'
```

11. Several results appear in the search. Look for the one with the name: **Core Networking Diagnostics - ICMP Echo Request (ICMPv4-In)**
12. This firewall rule object, starting from *Name* all the way down to *PackageFamilyName* lists all the configured properties for this rule.
13. Read through these properties. Recognize any from our GUI version?
14. This rule matches both our incoming ping requests and keeps to *Private*,

Public profiles. As with our GUI version, the scope is defaulted to local subnet only.

15. Let's turn on this incoming firewall rule by running the following command:

```
Enable-NetFirewallRule -Name CoreNet-Diag-ICMP4-EchoRequest-In
```

16. This selects the right rule and enables it.

17. One of the things you **must** get into the habit of doing with CLI commands is ***double-checking your work***.

18. Let's do that now by asking the system if it was actually enabled:

```
Get-NetFirewallRule -Name CoreNet-Diag-ICMP4-EchoRequest-In
```

19. Look for the *Enabled:* field. See how it's changed to **True**? It worked!

20. Optional: We can even check the scope for local subnet only as we did in the GUI version if we want with the following command:

```
Get-NetFirewallRule -Name CoreNet-Diag-ICMP4-EchoRequest-In | Get-NetFirewallAddressFilter
```

Output:

```
LocalAddress   : Any
RemoteAddress  : LocalSubnet4
```

21. Last, because there are two incoming ICMP rules (*Domain* profile and *Private, Public* profile), let's check that only the *Private, Public* rule is enabled:

Command:

```
Get-NetFirewallRule -DisplayName "Core Networking Diagnostics -  
ICMP Echo Request (ICMPv4-In)" | Select-Object DisplayName,  
Profile, Enabled
```

22. This shows a brief status of both rules. The *Domain* version should show **False** under the *Enabled* header, while the *Private*, *Public* version should show **True**.

Output:

```
DisplayName  
Profile Enabled  
-----  
-----  
Core Networking Diagnostics - ICMP Echo Request  
(ICMPv4-In)           Domain    False  
Core Networking Diagnostics - ICMP Echo Request (ICMPv4-In)  
Private, Public      True
```

23. Now, let's test our ping. Switch over to your Windows 11 Client (*client1*).
24. Open a Command Prompt window, and run the following command: `ping` `srv2-ipaddress` (Refer to *Part 3, Step 3*).
25. Does it work?
26. **If it does, congratulations!** You've just enabled ping for connectivity checking to *srv2* and gone through your first foray into the PowerShell!

Part 6: Windows Client (*client1*) - Applying Firewall Rules

Finally, let's enable ping on our Windows Client machine.

1. Login to *client1*.

2. Click the **Start** button and type the following search: **firewall**
3. Of the options that appear, select: **Windows Defender Firewall with Advanced Security**
4. Just as in *Part 2* with *srv1*, navigate to **Inbound Rules**.
5. Click the *Name* header to sort by name.
6. Find the following rule:
7. Name: **Core Networking Diagnostics - ICMP Echo Request (ICMPv4-In)**
8. Profile: **Private, Public**
9. Double-click it to open the rule's configuration settings.
10. In the *General* tab, find and check the box next to **Enabled**.
11. Click **Apply**, then **OK**.
12. Open a *Command Prompt* window and run the following to get your client's IP address: `ipconfig`
13. Switch to your *Server Core (srv2)* machine.
14. In PowerShell, run the following command: `ping client1-ipaddress` (Where client1-ipaddress is the address from Step 10.)
15. Example: `ping 10.0.40.11`
16. Does it work?
17. **It if does, congratulations!** You've just enabled ping for connectivity checking to *client1* and have full connectivity checking for your entire environment! This will become **very** handy in Labs 3-4.

Investigation 2: Remote Management - Windows Server GUI (srv1)

While you have direct access to all three VMs from VMware, these days most

machines are remote. This means you typically do not have direct, physical access to the servers.

Even if you do, having remote access to allow you to manage your servers from the comfort of your office (or home!) instead of walking to each physical machine is far better.

In this investigation, we'll set up remote access to our Windows Server GUI (srv1) so we can connect to it using your Windows 11 Client VM.

Part 1: Enabling Remote Desktop Connections

In this part, we'll turn on Remote Desktop (RDP) on the server. This will allow you to remotely connect to *srv1* from your Windows 11 Client.

1. In the *Server Manager* application, navigate to *Local Server*.
2. In the main *Properties* window, find the line entry for: **Remote Desktop**
3. Click the link next to it: **Disabled**
4. In the new *System Properties* window, you should automatically be on the *Remote* tab.
5. Look down to the section on **Remote Desktop**.
6. Toggle the option for: **Allow remote connections to this computer**
7. Ensure the checkbox next to this is on: **Allow connections only from computers running Remote Desktop with Network Level Authentication (recommended)**
8. Click **Apply**, then **OK**.

Part 2: Adding your RDP Firewall Rules

In this part, we'll add a firewall rule to allow the connection over the local network.

1. On *srv1*, open **Windows Defender Firewall with Advanced Security**.
2. In the *Inbound Rules* section, look for the following rules:
 - i. **Remote Desktop - Shadow (TCP-In)**
 - ii. **Remote Desktop - User Mode (TCP-In)**
 - iii. **Remote Desktop - User Mode (UDP-In)**
3. If they're all enabled (they should be), you're good to go!
4. Why check? Always check your firewall rules when enabling a new service, even if it *usually* enables the associated firewall rule automatically. **Never assume!**
5. If these rules are **not** enabled, enable them now.

Part 3: Connecting to Windows Server (srv1) from Windows 11

In this part, we'll verify our work by connecting to the server using our Windows 11 client VM.

1. Make sure you've got your *srv1* IP address from earlier. (You wrote it down, *right?*)
2. On *client1*, click the **Start** menu.
3. Search for and open: **Remote Desktop Connection**
4. In the newly opened application, in the *Computer* field, enter the IP address for *srv1*.
5. Click **Connect**.
6. Enter your *srv1* username and password, then click **OK**.
7. A security dialogue box pops up. **Check the box next to "*Don't ask me again for connections to this computer*" before the next step.**
8. Click **Yes**.
9. If you can see your *srv1* desktop, congratulations! You now have remote access.

10. To quit the remote session, find the floating HUD at the top of the screen with the IP address and click on the **X** icon.
11. Note: **This will not shutdown your *srv1* VM. It only ends your remote session.**
12. Note 2: A remote session will automatically lock your VM's direct session. You will need to unlock it again when you go back to it through VMware Workstation directly.

Investigation 3: Remote Management - Windows Server Core (*srv2*) with Remote Desktop

In this investigation, we'll set up remote access to our Windows Server Core (*srv2*) so we can connect to it using your Windows 11 Client VM.

Part 1: Enabling Remote Desktop Connections

In this part, we'll turn on Remote Desktop (RDP) on the Core server (*srv2*).

1. Login to *srv2*.
2. If you're at the PowerShell command prompt, run the following command to get back into the text-based *Server Manager* application: `sconfig`
3. Select option 7.
4. In the *Remote desktop* page, select option **E** to enable Remote Desktop.
5. It will ask you to select a security level. Select **Option 1**.
6. Press **Enter** to complete and go back to the main *Server Manager* page.
7. **Double-check your work.** Select option 7 again.
8. Is the *Remote desktop status* changed and correct?
9. If not, go through Steps 4-9 again.

10. If yes, press **Enter** without entering any options to go back to the main screen without making any changes.

Part 2: Adding a RDP Firewall Rules

In this part, we'll check that the RDP firewall rules have been enabled.

1. Select **Option 15** to exit to PowerShell.
2. Run the following PowerShell command to check the status of your RDP firewall rules:

Command:

```
Get-NetFirewallRule | Where-Object { $_.DisplayName -like '*Remote
Desktop*' -and $_.Direction -eq 'Inbound' } |
    Select-Object Name, DisplayName, Profile, Enabled
```

Output:

Name	DisplayName
Profile Enabled	
----	-----

RemoteDesktop-In-TCP-WS	Remote Desktop - (TCP-WS-
In) Any False	
RemoteDesktop-In-TCP-WSS	Remote Desktop - (TCP-WSS-
In) Any False	
RemoteDesktop-Shadow-In-TCP	Remote Desktop - Shadow (TCP-
In) Any True	
RemoteDesktop-UserMode-In-TCP	Remote Desktop - User Mode (TCP-
In) Any True	
RemoteDesktop-UserMode-In-UDP	Remote Desktop - User Mode (UDP-
In) Any True	

3. Check that your output matches the above. This is the same as on *srv1*, just at the command line. Notice which are enabled and which are not?
4. If you need to, grab the IP address for *srv2* here by running this command in PowerShell: `ipconfig`

Part 3: Connecting to Windows Server (*srv2*) from Windows 11 via RDP

In this part, we'll verify our work by connecting to *srv2* using our Windows 11 client VM. This is essentially the same as our instructions for *srv1*, but with a different IP address.

1. Make sure you've got your *srv2* IP address from earlier. (You wrote it down, *right?*)
2. On *client1*, click the **Start** menu.
3. Search for and open: **Remote Desktop Connection**
4. In the newly opened application, in the *Computer* field, enter the IP address for *srv2*.
5. Click **Connect**.
6. Enter your *srv2* username and password, then click **OK**.
7. A security dialogue box pops up. **Check the box next to "*Don't ask me again for connections to this computer*" before the next step.**
8. Click **Yes**.
9. If you can see your *srv2* Command Prompt window, congratulations! You now have remote access.
10. To quit the remote session, find the floating HUD at the top of the screen with the IP address and click on the **X** icon.
11. Note: **This will not shutdown your *srv2* VM. It only ends your remote session.**
12. Note 2: A remote session will automatically lock your VM's direct session.

You will need to unlock it again when you go back to it through VMware Workstation directly.

Investigation 4: Remote Management - Windows Server Core (srv2) with SSH

Part 1: Enabling SSH Connections

In this part, we'll turn on incoming SSH connections so we can connect to our server using Visual Studio Code from the Windows 11 client.

1. Login to your Server Core (*srv2*).
2. Exit the *Server Manager* program by selecting **Option 15**.
3. In PowerShell, run the following command to install the OpenSSH Server

```
Add-WindowsCapability -Online -Name OpenSSH.Server~~~~0.0.1.0
```

4. After it installs, we have to start the SSH service. Run the following:

```
Start-Service sshd
```

5. We now have to set the SSH service to start automatically every time the system boots. Run the following:

```
Set-Service -Name sshd -StartupType 'Automatic'
```

```
# Install OpenSSH Server
```

Part 2: Adding an SSH Firewall Rule

1. We'll now add an SSH rule. One already exists, but it's only attached to the *Private* profile and would not work for us.
2. Instead, we'll create a custom rule. Run the following command:

```
New-NetFirewallRule -Name sshd -DisplayName 'OpenSSH Server (sshd)'  
-Enabled True -Direction Inbound -Protocol TCP -Action Allow  
-LocalPort 22 -Profile Private,Public
```

3. This rule sets the following:
 - i. Name: sshd
 - ii. Display Name: OpenSSH Server (sshd)
 - iii. Enabled: True
 - iv. Profile: Private, Public
 - v. Direction: Inbound
 - vi. Action: Allow
 - vii. Protocol: TCP
 - viii. Port: 22
4. Verify your work! First, let's look at the firewall object by running the following command:
5. Now, let's verify the protocol and port with the following command:

Command:

```
Get-NetFirewallRule -Name 'sshd'
```

Output:


```

Name                : sshd
DisplayName          : OpenSSH Server (sshd)
Description          :
DisplayGroup        :
Group               :
Enabled             : True
Profile             : Private, Public
Platform            : {}
Direction           : Inbound
Action              : Allow
EdgeTraversalPolicy : Block
LooseSourceMapping  : False
LocalOnlyMapping    : False
Owner               :
PrimaryStatus       : OK
Status              : The rule was parsed successfully
                    from the store. (65536)
EnforcementStatus   : NotApplicable
PolicyStoreSource    : PersistentStore
PolicyStoreSourceType : Local
RemoteDynamicKeywordAddresses : {}
PolicyAppId         :
PackageFamilyName   :

```

Command:

```

Get-NetFirewallRule -Name 'sshd' | Get-NetFirewallPortFilter |
Select-Object Name, Protocol, LocalPort

```

Output:

```

Name Protocol LocalPort
-----
TCP      22

```

6. If your own output looks like the above, congratulations! If not, try your configuration commands again or ask for help.

Part 3: Connecting to Windows Server Core (srv2) from Windows 11 via SSH

1. On your Windows 11 Client (*client1*), open Command Prompt.
2. Enter the following command: `ssh Administrator@srv2ipaddress` (Where *srv2ipaddress* is your actual *srv2* IP address.)

Example: `ssh Administrator@10.0.40.2`

3. The first time you connect, a prompt will ask you if you are sure. Type **yes** and hit **Enter**.
4. Type your password when asked and hit **Enter**.

i Note: By default, You will *not* see asterisks or other characters as you type. **This is normal.** It is taking your keyboard input. It may take you a few tries to get used to it.

5. Once logged in, you are now on *srv2*'s Command Prompt environment.
6. To get back into PowerShell (which you should), run: `powershell`
7. It is **not** recommended to run the *sconfig* program over an SSH connection. If you need to use the *sconfig* / *Server Manager* text-based program, connect through your RDP connection in Part 1.

i A note about SSH and Server Manager:

Your SSH connection will allow you to run the *sconfig* Server Manager program and load it. However, many of it's functions fail over an SSH connection because it relies on background interactive processes that can't run on an SSH session. They must be run from the RDP connection, despite it *looking* the same.

For an example of this, run `sconfig` from your SSH session and try to run Windows Updates. If it finds any and you try to install them, it will fail. This is an SSH+sconfig problem; run the same thing through RDP and it will work fine.

Lab 3 Sign-Off

It's essential to complete Lab 3 correctly, as the networking setup will be required for all advanced labs.

When you finish Lab 3, ask your instructor for a sign-off.

Sign-Off Checklist

On *client1*:

1. Ping *srv1*
2. Ping *srv2*
3. SSH into *srv2*
4. RDP into *srv1*
5. In Firefox, load the IIS test page at: **10.0.10.1**
6. In Firefox, test your Internet connection by loading: **eff.org**

Lab 4 - Implementing DNS

Lab Preparation

Purpose / Objectives of Lab 4

In this lab, you will deploy and validate a local DNS service on your Windows Server GUI VM (*srv1*), then configure all machines in your lab environment to use it. By the end, you will be able to:

1. Install the **DNS Server** role on *srv1*.
2. Create a **Forward Lookup Zone** for *yourSenecaUsername.com*.
3. Add *A records* for *srv1*, *srv2*, *client1* and *client2*.
4. Configure forwarders so *srv1* can resolve Internet names.
5. Point *srv1* to itself via loopback (127.0.0.1) for DNS.
6. Reconfigure *srv2*, *client1*, and *client2* to use *srv1* as their DNS.
7. Create a **Reverse Lookup Zone** and **PTR records**. Verify forward and reverse resolution with `nslookup`, `ping`, and browser tests.

If you encounter technical issues, please contact your professor via e-mail or in your section's Microsoft Teams group.

Minimum Requirements

Before beginning, you must have:

1. Successfully completed Lab 3.

2. Attended the Week 5 lecture
3. Read through the Week 5 slides and have them handy as a reference for concepts.
4. Your external SSD (or approved personal computer) with your VMs from Labs 1-3.
5. Your VM login credentials.
6. Optional, but recommended: Caffeine delivery system.

Key Concepts

What DNS Does - Domain Name System

DNS maps names to IP addresses. Humans remember reddit.com. The Internet only understands how to route to 151.101.193.140. It's hierarchical and distributed, so no single server knows everything. **Authoritative servers** answer for their zones and resolvers figure the rest out.

Forward vs. Reverse Lookups

1. **Forward:** URL to IP (e.g., srv1.yourSenecaUsername.com converts to 10.0.UID.1) using A/AAAA records.
2. **Reverse:** IP to URL (e.g., 10.0.UID.1 to srv1.yourSenecaUsername.com) using PTR records in a Reverse Lookup Zone.

Reverse lookup is gold for troubleshooting and clean logs.

Zones and Authority

A zone is the authoritative dataset for a chunk of namespace. In this lab you create:

1. **Forward zone:** yourSenecaUsername.com
2. **Reverse zone:** 10.0.UID.0/24 (IPv4 reverse)

Each zone has an SOA (start of authority) and NS records defining who's in charge. We keep dynamic updates disabled here for control and repeatability.

Authoritative vs. Recursive (srv1 does both)

1. **Authoritative:** Answers with authority for names inside its zones.
2. **Recursive:** Chases down answers it doesn't know yet.

In our setup, *srv1* is **authoritative for your lab domain** and recursive for everything else (via forwarders).

Forwarders vs. Root Hints (and Conditional Forwarders)

1. **Forwarders:** send unknown queries to specific upstream resolvers (what we configure for the classroom).
2. **Root hints:** the built-in map to the DNS root—slower to “warm up.”
3. **Conditional forwarders:** send only certain domains to specific DNS servers (handy in enterprises; not needed here).

We will only be using forwarders in this lab for ease of set up.

Records You'll Actually Touch

1. A: Hostname to IPv4.
2. PTR: IPv4 address to hostname (reverse).
3. CNAME (alias): points a name to another name.
4. Rule of thumb: a CNAME can't live alongside other records at the same name (no A+MX+CNAME pileups).

FQDNs and the DNS Suffix

1. Fully-Qualified Domain Name (FQDN): Full path (e.g., *srv1.yourname.com.*).
2. **Short names** (e.g., ping *srv1*) rely on a DNS suffix search list. To keep tests deterministic, we use FQDNs in this lab.

Caching and TTL

DNS answers are cached to reduce load. The TTL controls how long.

Tips:

1. After changing records, clear caches where relevant:
2. **Client:** `ipconfig /flushdns`
3. **Server:** DNS Manager > right-click server > Clear Cache

Ports and Firewall

DNS uses **UDP/53** for most queries, **TCP/53** for large responses and zone transfers. With the DNS Server Role installed, inbound DNS rules should be enabled (Private/Public).

Verify. Never assume.

Loopback and Why We Use It on *srv1*

The loopback interface (127.0.0.1) sends traffic to the local host. It is immune to network interface card (NIC) misconfigurations. Pointing *srv1* at **127.0.0.1** ensures it always asks itself first, which simplifies troubleshooting.

```
Loop: srv1 > 127.0.0.1 > srv1
```

Split-Horizon (a.k.a. Split-Brain) DNS (just awareness)

Enterprises often serve different answers internally VS externally for the same name. In this lab your domain is internal-only. Don't register it publicly.

Troubleshooting Toolkit

1. **nslookup** (classic): Quick forward/reverse tests.

2. **Resolve-DnsName** (PowerShell): Richer diagnostics (optional).
3. **Get-DnsClientServerAddress**: Confirm which DNS your client is using.
4. If names resolve but pings fail, think firewall (Lab 3), not DNS.

Investigation 1: Configuring DNS on Windows Server (*srv1*)

Part 1: Confirm a Static IP Address on *srv1*

As *srv1* will be our primary network services provider, we have to give it a **static IP address**. This is so it doesn't periodically change. If it did change, any other machines relying on this one would suddenly be unable to connect and lose access to all services provided, including the very DNS we're about to set up!

We already did this during *Lab 2, Investigation 1, Part 3*. Doing so is called **manual static network configuration**.

Let's confirm our work:

1. Open **Command Prompt**.
2. Run the following command: `ipconfig`
3. Confirm the *Internal Network* adapter uses: **10.0.0.1**
4. Find the *External Network* adapter's current DNS address. **Write this down**. You will need it later in this lab.

Static IP addressing is always required for a computer or device providing network services. Remember that.

If the proper static IP is confirmed, move on to *Part 2*.

Part 2: Installing the DNS Server Role on *srv1*

1. **Login to *srv1*** with your Administrator account.
2. Open the **Server Manager** application (icon is on the taskbar or Start menu).
3. In the left menu, click on **Local Server** if you're not already there.
4. At the top right, click **Manage > Add Roles and Features**.
5. In the **Add Roles and Features Wizard** window:
 - i. On *Installation Type*, select **Role-based or feature-based installation**, then **Next**.
 - ii. On *Server Selection*, confirm **Select a server from the server pool** and choose ***srv1***, then **Next**.
 - iii. On *Server Roles*, scroll down and check the box for **DNS Server**.
 - iv. A pop-up will ask to add required features, click **Add Features**.
 - v. Back on the *Server Roles* page, click **Next**.
 - vi. On the *Features* page, leave defaults, click **Next**.
 - vii. On the *DNS Server* page, read and click **Next**.
 - viii. On the *Confirmation* page, click **Install**.
 - ix. Wait for installation to finish (may take a few minutes).
 - x. When you see *Installation succeeded on *srv1**, click **Close**.

Part 3: Configuring a New DNS Zone

1. Back in **Server Manager**, in the top right, click **Tools > DNS** to open the DNS Manager.
2. In the left pane, expand ***srv1*** and then **Forward Lookup Zones**.
3. Right-click on **Forward Lookup Zones** and select **New Zone...**
4. In the **New Zone Wizard**:

- i. On *Zone Type*, select **Primary zone**, click **Next**.
- ii. On *Zone Name*, enter ***yourSenecaUsername.com***, replacing *yourSenecaUsername* with your actual username.
- iii. For example, mine might be: **cjohnson30.com**
- iv. Click **Next**.
- v. On *Zone File*, keep **Create a new file** selected, click **Next**.
- vi. On *Dynamic Update*, keep **Do not allow dynamic updates** (default), then click **Next**.
- vii. Click **Finish** to create the zone.

Part 4: Adding Host (A) Records

1. In **DNS Manager**, expand your new ***yourSenecaUsername.com*** zone.
2. Right-click in the right pane and select **New Host (A or AAAA)...**
3. For each of your lab VMs, create a record:
 - i. **srv1**
 - Name:
 - IP address:
 - Click **Add Host**
 - ii. **srv2**
 - Name:
 - IP address:
 - Click **Add Host**
 - iii. **client1**
 - Name:
 - IP address:
 - Click **Add Host**
 - iv. **client2**
 - Name:

- IP address: (Use the IPv4 address of client2)
- Click **Add Host**

4. When finished, you should see all four names listed in the zone.

Part 5: Configure DNS Forwarders on *srv1*

We want *srv1* to answer local names (your zone which contains *srv1*, *srv2*, etc.) and resolve Internet names. If *srv1* isn't authoritative for a name (i.e. doesn't have a definition for it), it *should* forward the query upstream.

At the moment, it's only set up to give the IP addresses of our VMs, not the Internet.

To fix this, we have to tell *this* server where to look if it doesn't have the answer locally. If it can't answer locally, it will forward the request to an Internet-based DNS server.

This is called a *Forwarder*. Let's set it up:

1. In **DNS Manager** on *srv1*, right-click on your server's name in the left pane.
2. Select **Properties**.
3. Go to the **Forwarders** tab.
4. Click **Edit...** Investigation 1, Part 1, Step 4.
5. In the *IP address* field, enter the IP address from *Investigation 1, Part 1, Step 4*, then hit **Enter** for it to be verified. If it can reach it, you should see a domain name populate next to the IP address.
6. Add a second DNS for fallback, using CIRA's DNS:
 - **149.112.121.20**
7. Make sure CIRA's DNS is the **second** entry in the list.
8. Click **OK** to add.

9. Click **OK** again to close Properties.

i Note: In managed networks you might use *Root Hints* instead. We're using Forwarders for predictable performance in a classroom environment.

Part 6: Configuring *srv1* To Use DNS From *srv1*

Now that we've set up our DNS server locally on *srv1*, we have to tell *srv1* to use itself for DNS resolution instead of what Seneca (or your ISP) provides.

We're going to do this by using the *Loopback Interface*. This is a special network interface on all modern operating systems that, when network traffic (like DNS!) is sent to it, will loop it back to itself. Hence, the name.

The *Loopback* interface's address is always **127.0.0.1** on all systems. (You may sometimes see **localhost** used for this. That's the URL version of the exact same network interface.)

This special interface is great for a number of reasons, but most importantly: It's completely separate from our other networks and so isn't affected by network configuration mistakes. When you use loopback, you know you're getting the most direct link. Helpful in troubleshooting!

Network Flow with Loopback: Send Traffic from *srv1* > *srv1* Loopback > Receive Traffic from *srv1*

1. To do this, open up the Network Properties for **External Network** and change its TCP/IPv4 settings to the following:
 - Preferred DNS server: **127.0.0.1**
 - Alternate DNS server:
2. Click OK on each open dialog box until you're back at *Network*

Connections.

3. Repeat Steps 1-2 for **vEthernet (HQ Network)**.
4. Verify your work by opening **Command Prompt** and running: `ipconfig /all`
5. You should see both interfaces and their DNS settings set to 127.0.0.1.

If you do, move on to the next part!

Part 7: Test *srv1* DNS Resolution

Our first task is to confirm this works locally by telling the server to talk to itself first for DNS resolution. This lets us test our DNS service and records while removing the network and firewall pieces. (See the explanation above in Part 6.)

1. Open **Command Prompt**.
2. To test DNS resolution functionality on your local network, run the following commands one at a time:

```
nslookup srv1.cjohnson30.com
nslookup srv2.cjohnson30.com
nslookup client1.cjohnson30.com
nslookup client2.cjohnson30.com
```

3. Now, let's run a lookup for an address on the greater Internet:

```
nslookup eff.org
nslookup reddit.com
```

4. If each returns the proper IP address value, you're done! If not, revisit your earlier configuration steps and check to see what might be missing.
5. Don't be afraid to ask for help!
6. **If *any* of your lookups above failed, ask for help! Do not move onto the next Investigation if failed.**

Investigation 2: Configuring Other Machines For DNS Use

In this investigation, we will configure our other VMs to use *srv1* as the DNS server for all DNS resolution.

i A note about the *loopback* address from Investigation 1

The *loopback address* we used on *srv1* only works locally. You cannot use this address outside of *srv1*. You must use its **Internal Network** address.

Part 1: Windows Server Core (srv2)

1. Turn on the following VMs:
 - *srv2*
 - *client1*
 - *client2*
2. On *srv2*, log in as Administrator.
3. At the prompt, run:

```
Get-NetIPAddress
```

3. Look for the section that contains *srv2*'s normal **Internal Network** IP address. Make a note of the **InterfaceIndex** number just below it. (For

example, 4. Yours may differ.)

4. Set the DNS server to srv1's IP (replace X.X.X.X with srv1's actual address, and 4 with your NIC's ifIndex):

```
Set-DnsClientServerAddress -InterfaceIndex 4 -ServerAddresses  
`10.0.0.1`
```

5. Confirm the change:

```
Get-DnsClientServerAddress
```

6. Repeat Steps 2-5 for the **External Network** address/network interface.
7. When done, the command on Step 5 should provide a similar output to this:



```
Administrator: C:\WINDOWS\system32\cmd.exe  
PS C:\Users\Administrator> Get-DnsClientServerAddress  
  
InterfaceAlias      Interface Index Address Family ServerAddresses  
-----  
External Network    4 IPv4 {10.0.45.1}  
External Network    4 IPv6 {}  
Internal Network    7 IPv4 {10.0.45.1}  
Internal Network    7 IPv6 {}  
Loopback Pseudo-Interface 1 1 IPv4 {}  
Loopback Pseudo-Interface 1 1 IPv6 {fec0:0:0:ffff::1,  
  
PS C:\Users\Administrator> _
```

8. To test DNS resolution functionality, run the following commands one at a

time:

```
nslookup srv1.cjohnson30.com
nslookup srv2.cjohnson30.com
nslookup client1.cjohnson30.com
nslookup client2.cjohnson30.com
nslookup eff.org
```

7. If each returns the proper IP address value, you're done! If not, revisit your earlier configuration steps and check to see what might be missing.
8. Don't be afraid to ask for help!
9. Last, ping each other machine to prove you no longer need to remember IP addresses:

```
ping srv1.cjohnson30.com
ping client1.cjohnson30.com
ping client2.cjohnson30.com
```

13. Cool, right?

Part 2: Windows 11 Client (client1)

1. Login as Administrator.
2. Right-click the **Network** icon in the system tray, click **Network and Internet settings**.
3. Click **Ethernet**.
4. Scroll down to *DNS server assignment* and click the **Edit** button next to it.
5. In the *Edit DNS settings* pop-up window, click the drop-down menu to change it from *Automatic (DHCP)* to **Manual**.
6. Find the new *IPv4* option and toggle it **On**.

7. In the *Preferred DNS* field, enter the *srv1* IP address.
8. Leave all others on their defaults and click **Save**.
9. Remove saved DNS cache. Open **Command Prompt** and type:

```
ipconfig /flushdns
```

9. To test DNS resolution functionality, run the following commands one at a time:

```
nslookup srv1.cjohnson30.com  
nslookup srv2.cjohnson30.com  
nslookup client1.cjohnson30.com  
nslookup client2.cjohnson30.com  
nslookup eff.org  
nslookup reddit.com
```

10. If each returns the proper IP address value, you're done! If not, revisit your earlier configuration steps and check to see what might be missing.
11. Don't be afraid to ask for help!
12. Last, ping each server to prove you no longer need to remember IP addresses:

```
ping srv1.cjohnson30.com  
ping srv2.cjohnson30.com  
ping client2.cjohnson30.com
```

13. In Firefox, browse to your *srv1* IIS splash page by name:

```
http://srv1.cjohnson30.com
```

14. Also cool, right?

Investigation 3: Configuring Reverse Lookup (PTR) Records

In this investigation, we will set up Reverse Lookup (PTR) records so we can look up a server by its IP address and get a FQDN address back. Very helpful in troubleshooting and a best practice.

You'll also see how it helps for self-identification a little later.

Part 1: Create a Reverse Lookup Zone

1. In **DNS Manager** on *srv1*, right-click **Reverse Lookup Zones** in the left pane.
2. Click **New Zone...**
3. In the **New Zone Wizard**:
 - i. On *Zone Type*, select **Primary zone**. Click **Next**.
 - ii. On *Reverse Lookup Zone Name*, choose **IPv4 Reverse Lookup Zone**. Click **Next**.
 - iii. On *Network ID*, enter the first three octets of your VMs' subnet (for example, if your VMs' IPs are 10.0.45.x, enter 10.0.45). Click **Next**.
 - iv. On *Zone File*, keep defaults. Click **Next**.
 - v. On *Dynamic Update*, keep defaults (Do not allow dynamic updates). Click **Next**.
 - vi. Click **Finish** to create the zone.

Part 2: Add PTR Records for Each VM

1. We're going to automatically add PTR records using our existing A records for ease of use.
2. Inside the *Forward Lookup Zones*, in the **cjohnson30.com** zone, right-click each A record you created (srv1, srv2, client1, client2) and select **Properties**.
3. Check the box next to **“Update associated pointer (PTR) record”** and click **OK**.
4. If the box was already checked, uncheck it, click **Apply**, check it again, then click **Apply** again. This forces the PTR generation.
5. This should automatically create PTR records in your new reverse zone.
6. Double-check your work. Click on the folder inside *Reverse Lookup Zone* to see its contents. If empty, right-click on the folder and click *Refresh*.
7. They should now appear. If they do not, retrace your steps or ask for help.

Part 3: Test Reverse DNS Resolution

1. On *srv2*, open PowerShell.
2. Run **nslookup** on each VM's IP address:

```
nslookup 10.0.UID.1
nslookup 10.0.UID.2
nslookup 10.0.UID.11
nslookup 10.0.UID.12
```

3. You should see the PTR record and corresponding hostname for each IP address.

Example Output:

```
Server:  srv1.cjohnson30.com  
Address: 10.0.45.1
```

```
Server:  srv2.cjohnson30.com  
Address: 10.0.45.2
```

```
Name:    client1.cjohnson30.com  
Address: 10.0.45.11
```

```
Name:    client2.cjohnson30.com  
Address: 10.0.45.12
```

4. Notice that the server entry no longer says "UnKnown" as it did before. Now it can identify itself!

Congratulations, you've now set up basic DNS services from *srv1* to each VM in your environment and allowed for local resolution to your other VMs by name!

Troubleshooting and Common Pitfalls

1. **Forwarders not working:** Verify upstream DNS IPs are reachable from *srv1* and listed in the Forwarders tab (not just in adapter DNS settings).
2. **Wrong adapter DNS:** On *srv1*, both adapters should use 127.0.0.1. On other machines, adapters should use 10.0.UID.1.
3. **Names resolve, but pings fail:** That's likely firewall (Lab 3). The important bit for this lab is that `nslookup` returns correct answers.
4. **PTRs missing:** Re-toggle "Update associated PTR" on each A record, Apply, then Refresh the reverse zone.
5. **Typo domain:** Be consistent (e.g., *cjohnson30.com* everywhere).
6. **UID mismatch:** Ensure all A/PTR records and tests use your UID subnet

(10.0.UID.x).

Lab 4 Sign-Off

It's essential to complete Lab 4 correctly. All later labs assume working forward and reverse name resolution.

When you finish Lab 4, ask your instructor for a sign-off.

Sign-Off Checklist

Please have the following on screen and ready to show.

On srv1:

1. `nslookup srv1.yourdomain`: Returns 10.0.UID.1
2. `nslookup eff.org`: Returns a public IP (forwarder working)
3. Both adapters show 127.0.0.1 as DNS in `ipconfig /all`
4. Your **Forward Lookup Zone** page open for review.
5. Your **PTR Records** page open for review.

On srv2:

1. `Get-DnsClientServerAddress`: Shows 10.0.UID.1 for Internal and External network adapters.
2. `nslookup client1.yourdomain` and `nslookup client2.yourdomain`: Correct IPs
3. `nslookup 10.0.UID.1`, `nslookup 10.0.UID.11` and `nslookup 10.0.UID.12`: Correct PTR to FQDN.

On client1:

1. DNS set to 10.0.UID.1 (Manual/IPv4)
2. `nslookup srv1.yourSenecaUsername.com`: Resolves to 10.0.UID.1

3. In Firefox: `http://srv1.yourSenecaUsername.com` loads the IIS splash page.
4. `nslookup reddit.com`: Resolves a public IP (forwarder working)
5. In Firefox: `http://www.reddit.com` loads the Reddit main page.

Lab 5 - Implementing DHCP

Lab Preparation

Purpose / Objectives of Lab 5

In this lab, you will deploy **DHCP** on *srv1* to automatically provide IP configuration for your internal lab network.

By the end, you will:

1. Install the **DHCP Server** role on *srv1*.
2. Create an IPv4 **scope** for your internal subnet **10.0.UID.0/24** with proper **options** (Router, DNS servers, DNS Suffix).
3. Create **reservations** so *srv2*, *client1*, and *client2* receive predictable addresses that match prior labs.
4. Flip *srv2*, *client1*, and *client2* from **static** addressing to **DHCP**, then verify leases.
5. Validate **DHCP + DNS** end-to-end with `ipconfig`, `nslookup`, and `ping`.

Minimum Requirements

Before beginning, you must have:

- Successfully completed **Lab 4 - Implementing DNS**, with *srv1* resolving internal names and using loopback for local DNS testing.
- Your **UID** value handy.

- All four VMs present from earlier labs: *srv1* (GUI), *srv2* (Core), *client1*, *client2*.

Key Concepts

- **DHCP** automatically assigns IP addressing (IPv4 address, subnet mask), **Default Gateway (Option 003)**, **DNS servers (Option 006)**, and **DNS Suffix (Option 015)** to clients on a subnet.
- **Scope** defines the usable address pool on a network. **Reservations** ensure specific MAC addresses always receive the same IP.
- In this course, your internal network remains **10.0.UID.0/24**; *srv1* is the router at **10.0.UID.1**.

Design Note: We will keep client addresses consistent with Labs 2–4 using **reservations** so your existing DNS host records remain correct.

Investigation 1: Install the DHCP Server Role on *srv1*

Part 1: Add the DHCP Server Role

1. **Login to *srv1*** with your Administrator account.
2. Open **Server Manager** → **Manage** > **Add Roles and Features**.
3. **Installation Type:** *Role-based or feature-based installation* → **Next**.
4. **Server Selection:** choose ***srv1*** → **Next**.
5. **Server Roles:** check **DHCP Server** → **Add Features** if prompted → **Next** through *Features* and *DHCP Server* pages.
6. **Confirmation:** **Install** and wait for completion.

Part 2: Post-Install Configuration

1. In **Server Manager**, click the yellow ! → **Complete DHCP configuration**.
 2. In the wizard, click **Commit** → **Close**.
-

Investigation 2: Create Scope and Options (10.0.UID.0/24)

Part 1: Create a New IPv4 Scope

1. **Tools > DHCP**.
2. Expand your server (srv1) → **IPv4** → **Right-click > New Scope...**
3. **Scope Name:** OSM620 HQ
4. **IP Address Range:**
 - **Start:** 10.0.UID.2
 - **End:** 10.0.UID.199
 - **Subnet mask:** 255.255.255.0
5. **Add Exclusions and Delay:** leave blank (we will use **Reservations** for fixed addresses).
6. **Lease Duration:** default (8 days) → **Next**.
7. **Configure DHCP Options:** choose **Yes, I want to configure these options now** → **Next**.

Part 2: Configure Scope Options

1. **Router (Default Gateway) - Option 003:** `10.0.UID.1` → **Add** → **Next**.
2. **Domain Name and DNS Servers - Option 006/015:**
 - **Parent domain** (Option 015): `yourSenecaUsername.com` (*replace with your lab domain from Lab 4*)
 - **DNS Servers** (Option 006): `10.0.UID.1` (*srv1*) → **Add** → **Next**.
3. **WINS Servers:** leave blank → **Next**.
4. **Activate Scope:** select **Yes, I want to activate this scope now** → **Finish**.

Why start at .2? .1 is the gateway on *srv1*. We'll reserve .2, .11, and .12 for *srv2*, *client1*, and *client2* to preserve continuity with earlier labs.

Investigation 3: Reservations for Predictable Addresses

We'll bind specific IPs to each machine's **MAC address** so they keep the same addresses used in Labs 2-4.

Part 1: Collect MAC addresses

1. On **srv2** (Core): `ipconfig /all` → record the **Physical Address** for the **Internal** adapter.
2. On **client1**: `ipconfig /all` → record Physical Address.

3. On **client2**: `ipconfig /all` → record Physical Address.

Part 2: Create reservations on *srv1*

In **DHCP Manager** → **IPv4** > **OSM620 HQ** > **Reservations** → **Right-click** > **New Reservation...**

Create the following (replace `UID` and MACs):

- **srv2**
 - **IP address:** `10.0.UID.2`
 - **MAC:** `<srv2 MAC>`
 - **Description:** `Windows Server 2025 Datacenter Core`
 - **Supported types:** **Both** → **Add**
- **client1**
 - **IP address:** `10.0.UID.11`
 - **MAC:** `<client1 MAC>`
 - **Description:** `Windows 11 Education`
 - **Supported types:** **Both** → **Add**
- **client2**
 - **IP address:** `10.0.UID.12`
 - **MAC:** `<client2 MAC>`
 - **Description:** `Windows 11 Education`
 - **Supported types:** **Both** → **Add**

Verify all three appear under **Reservations**.

Investigation 4: Switch from Static to DHCP

Part 1: *srv2* (Server Core)

1. Login as **Administrator**.
2. Identify interface index:

```
Get-NetIPInterface
```

3. Enable DHCP for IPv4 (replace **4** with your **InterfaceIndex** on the Internal adapter):

```
Set-NetIPInterface -InterfaceIndex 4 -Dhcp Enabled  
Set-DnsClientServerAddress -InterfaceIndex 4  
-ResetServerAddresses  
ipconfig /release  
ipconfig /renew  
ipconfig /all
```

4. Confirm **IPv4 Address = 10.0.UID.2, Default Gateway = 10.0.UID.1, DNS = 10.0.UID.1**.

Part 2: *client1* (Windows 11)

1. **Settings > Network & Internet > Ethernet > Edit** next to **IP assignment** → **Automatic (DHCP)** → **Save**.

2. **Edit** next to **DNS server assignment** → **Automatic (DHCP)** → **Save**.

3. Command Prompt:

```
ipconfig /release  
ipconfig /renew  
ipconfig /all
```

4. Confirm **IPv4 = 10.0.UID.11**, **Gateway 10.0.UID.1**, **DNS 10.0.UID.1**.

Part 3: *client2* (Windows 11)

Repeat the exact steps as client1. Confirm **IPv4 = 10.0.UID.12**.

If a machine doesn't pick up the reserved address, re-check the MAC you entered in the reservation and ensure the NIC you changed to DHCP is the **Internal** (HQ) adapter.

Investigation 5: Verify Leases and Name Resolution

Part 1: Check Leases in DHCP Manager

1. On *srv1*: **DHCP Manager** → **IPv4** > **OSM620 HQ** > **Address Leases**.
2. Confirm entries for **srv2**, **client1**, **client2** with correct **Reservation** type and IPs.

Part 2: Test DHCP + DNS

From **client1** (or **client2**):

```
ipconfig /all
nslookup srv1.yourSenecaUsername.com
nslookup client2.yourSenecaUsername.com
nslookup eff.org
ping srv1.yourSenecaUsername.com
```

- All should resolve and ping (ICMP allowed per Lab 3). If `nslookup` for Internet names fails, revisit **DNS Forwarders** on *srv1*.
-

Troubleshooting

- **Client keeps old static IP:** ensure **IP assignment = Automatic (DHCP)**; on Core, run `Set-NetIPInterface -Dhcp Enabled` then `ipconfig /release` + `/renew`.
 - **Gets wrong pool address:** verify the **reservation MAC** matches the **Internal adapter**; remove/re-add reservation if needed.
 - **No default gateway:** re-check **Option 003** in scope options.
 - **DNS wrong:** verify **Option 006** points to `10.0.UID.1` and **Option 015** is your lab domain.
 - **No Internet:** confirm *srv1* still NATs traffic between Internal and External (Lab 2 RRAS) and that **Forwarders** work (Lab 4).
-

Lab 5 Sign-Off

To complete Lab 5, show your instructor:

1. **Scope** `10.0.UID.2-199` active with Options **003**, **006**, **015** set.
2. **Reservations** for **srv2 (10.0.UID.2)**, **client1 (10.0.UID.11)**, **client2 (10.0.UID.12)**.
3. On each client, `ipconfig /all` showing **DHCP Enabled = Yes** and the correct leased address.
4. In Address Leases, entries for all three machines marked as **Reservation**.
5. `nslookup` for `srv1.yourSenecaUsername.com` **and** an Internet site returns addresses successfully.

You're done. Your lab enterprise now uses **DHCP** correctly while keeping earlier labs' addressing and DNS intact.

Assignment 1

To be released on Week 6.

Assignment 2

To be released on Week 9.