Lab 1: Setting Up The Environment

Server System Management - Windows Server Labs

<STUDENT NAME>

Academic Year: 2022-2023

Table of Contents

[Introduction 3](#_Toc127274973)

[Learning Goals 3](#_Toc127274974)

[Knowledge (what you need to know) 3](#_Toc127274975)

[Abilities (what you need to be able to do) 3](#_Toc127274976)

[Requirements 3](#_Toc127274977)

[Windows GUI Server 4](#_Toc127274978)

[Windows Core Server 5](#_Toc127274979)

[Windows 11 Client 6](#_Toc127274980)

## Introduction

In this lab you will install a basic Windows network environment, consisting of 2 Windows Servers (a full GUI server and a Core server) and a Windows 11 client. You’ll implement these different hosts as virtual machines, working in a virtual network, which means that everything will run on your own laptop/desktop computer. Finally, you’ll also install and explore some additional tools that we’ll be using throughout the course.

## Learning Goals

# Knowledge (what you need to know)

1. Know the differences and similarities between the different Windows server options
2. Know what a virtual machine (in VMware Workstation) is and how to navigate its files on the host system.
3. Know and understand the different execution policies of PowerShell and the defaults.

# Abilities (what you need to be able to do)

1. Create and install Windows server and Windows 11 client in virtual machines
2. Make sure the virtual machines are in the same (virtual) network
3. Install the VMware tools on the virtual machines
4. Create a local (offline) user account on the Windows 11 client.
5. Ensure Windows updates don’t start automatically
6. Run Pester tests
7. Use the following tools
   1. sconfig
   2. regedit
   3. gpedit
   4. windows terminal

## Requirements

1. VMware Workstation Pro (or other virtualization software)
2. Open the Howest TI NAS (first connect to Howest through the VPN if you are not on-campus):
   1. In the browser: <http://nas.ti.howest.be/> (login with firstname.lastname (not e-mail!) and your password)
   2. In file explorer: [\\nas.ti.howest.be](file://nas.ti.howest.be)
3. Browse to \TI-StudentShare\TI-S4-WindowServer and download both ISO files (One for Windows 11 and one for Windows Server).

## Windows GUI Server

1. Create a new virtual machine in VMware Workstation
   1. Choose **custom (advanced)** in the wizard for the configuration type.
   2. Hardware compatibility: **Workstation 16.2.x (default)**
   3. Choose: **“I will install the operating system later” (default)**
   4. Guest OS: **Windows Server 2019**
   5. Name the VM: “**Windows GUI Server 2022**” and change the location according to your preferences but do **not** put it in a cloud-synchronized folder such as Dropbox or Onedrive!
   6. Firmware type: **UEFI (default)** (leave secure boot unchecked)
   7. Processor: **default** is fine, more is optional depending on your hardware.
   8. Memory: **default** is fine, more is optional depending on your hardware.
   9. Network: **default** is fine – take note of this network option!

NAT seems to be the default

* 1. I/O controller type: **default** is fine

LSI Logic SAS

* 1. Disk type: **default** is fine

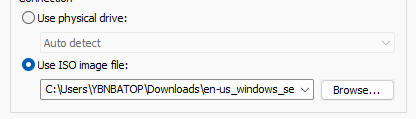
NVMe seems to be the chosen one

* 1. Disk: **create a new virtual disk** and **Store it as a single file**. Leave allocate all disk space unchecked (default)!
  2. Finish the wizard

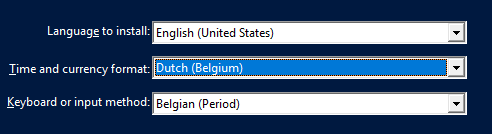
1. When the wizard closes, a new tab in VMware Workstation should open with the details of the new virtual machine. At the bottom (VM details) there is a line: **Configuration file.** Browse and verify the location before continuing the next step. Are you sure this is the correct place you want to store the virtual machine? How big is the vmdk-file now?

I used the default path for my installation, hence I think the configuration file will pop-up after the boot up and mount of the file

1. Go back to VMware Workstation and select the option “**edit virtual machine settings**”. Go to CD/DVD and select to use the ISO-file for Windows Server 2022 that you downloaded earlier. Make sure the ISO file is connected when the VM starts.



1. Start the virtual machine and press a button to boot from the ISO file to start the installation wizard.
   1. Select Dutch (Belgium) as “Time and Currency format” and select the keyboard layout that corresponds to the keyboard of your host laptop/pc. For Belgian (Azerty) configurations choose the following:

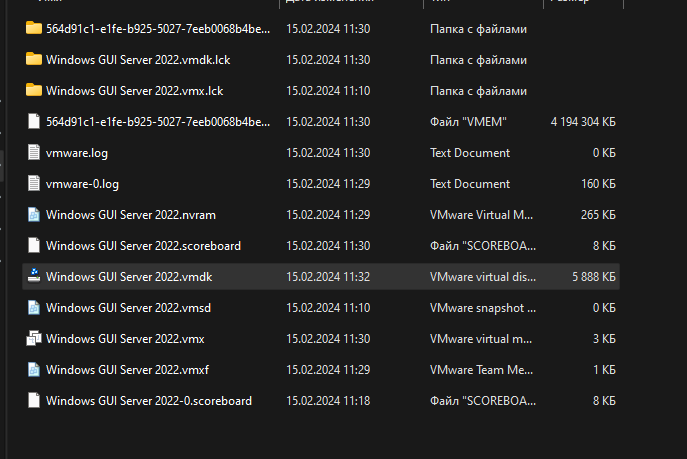


* 1. Press “install now”
  2. **Don’t enter a product key** but choose “**I don’t have a product key now”** instead.
  3. Choose **Windows Server 2022 Datacenter (Desktop Experience)**
  4. Accept the license
  5. Choose **Custom: Install Windows Only**
  6. Choose **Drive 0** to install Windows on.
  7. Choose an Administrator password (suggestion: Friday13th!) **Attention**: NumLock is not on in most cases.

I did use the password **Friday13th!** For future reference

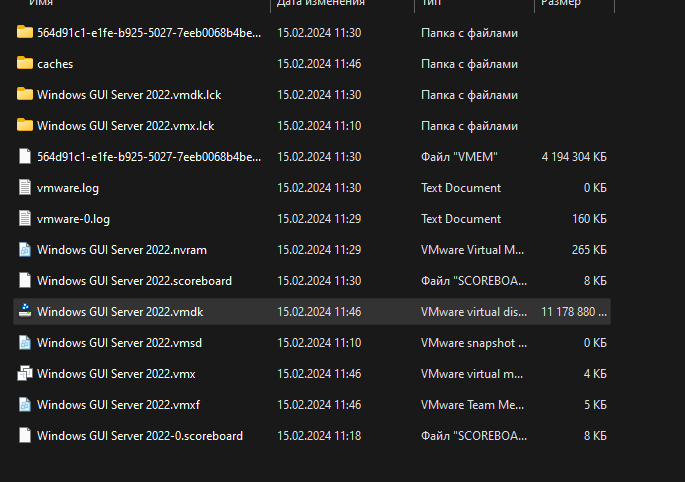
* 1. Finish

1. Once again browse to the location of your virtual machine configuration files, and take a look at the vmdk-file. How big is it now?

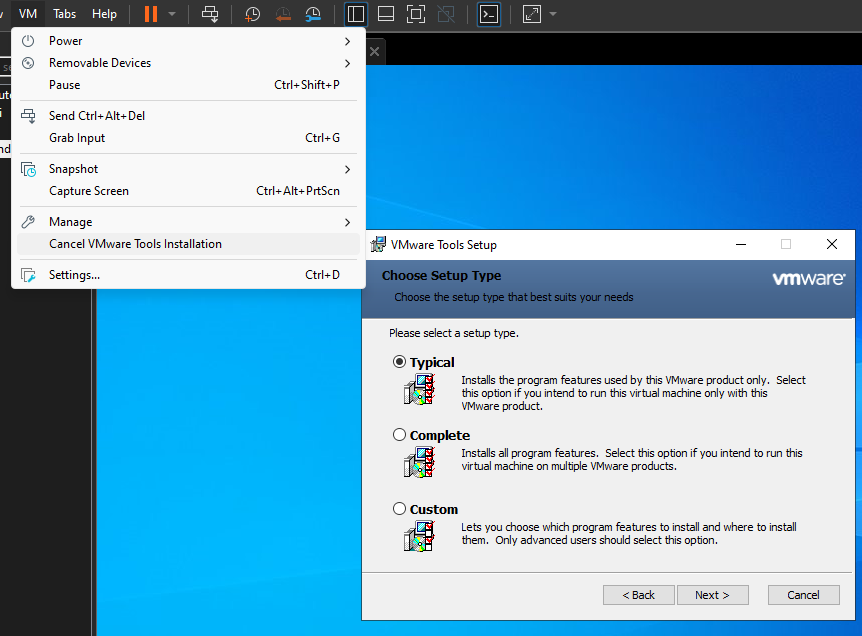


Does not seem to have changed much since before the install of the ISO.

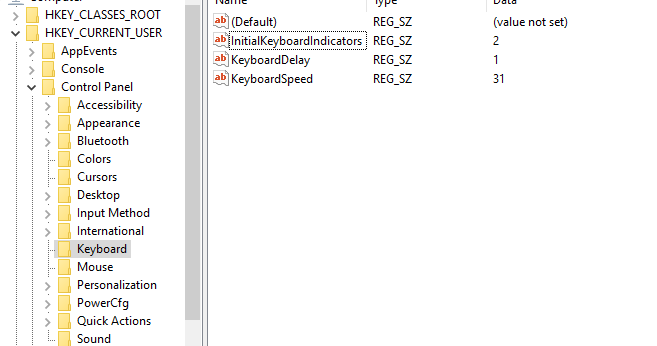
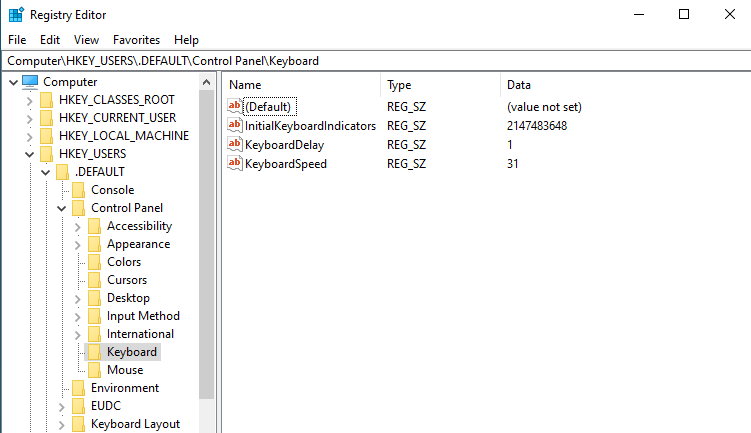
UPD: After the initialisation it has changed to a more storage space



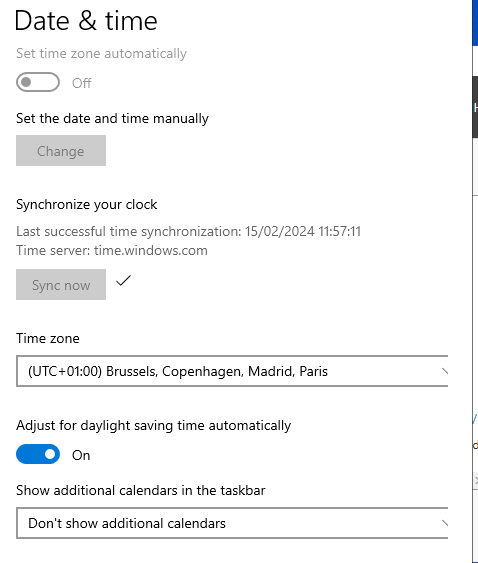
1. Sign in as Administrator on the Windows Server machine (TIP: CTRL+ALT+INSERT), allow the server to be discoverable on the network and install the VMware tools by pressing the button at the bottom right or via the Menu bar > VM > Install VMware tools… Note: VMware inserts a virtual DVD to the virtual machine. If you missed the popup, use file explorer to run the setup manually. Choose a “**typical install**”. You have to restart but you should see visual changes immediately.



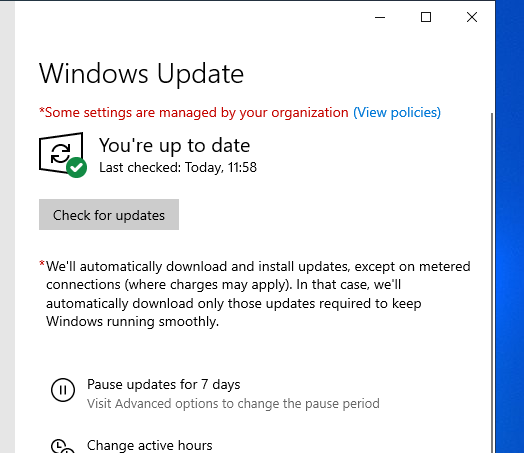
1. After restarting the virtual machine log in as Administrator. You might have noticed that once again NumLock is not on by default. Change this by editing the register. **Change “initialKeyboardIndicators” to “2”.** Once for .DEFAULT under HKEY\_USERS and once for HKEY\_CURRENT\_USER. You can find the InitialKeyboardIndicators under \Control Panel\Keyboard.



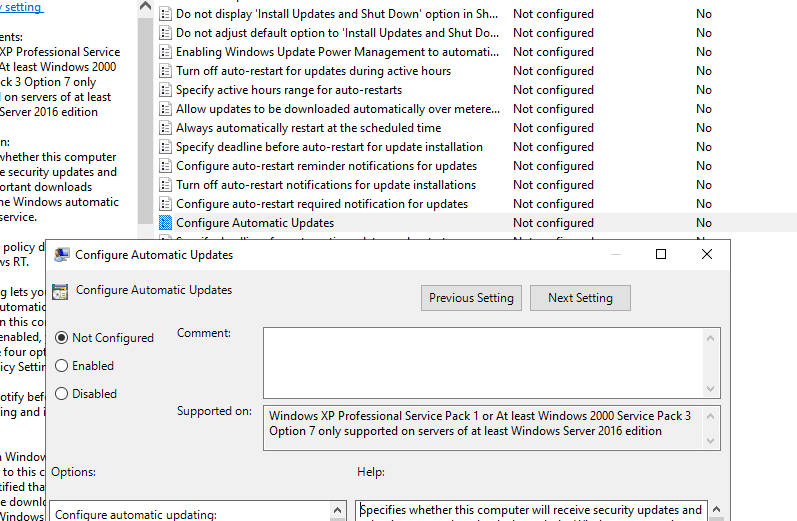
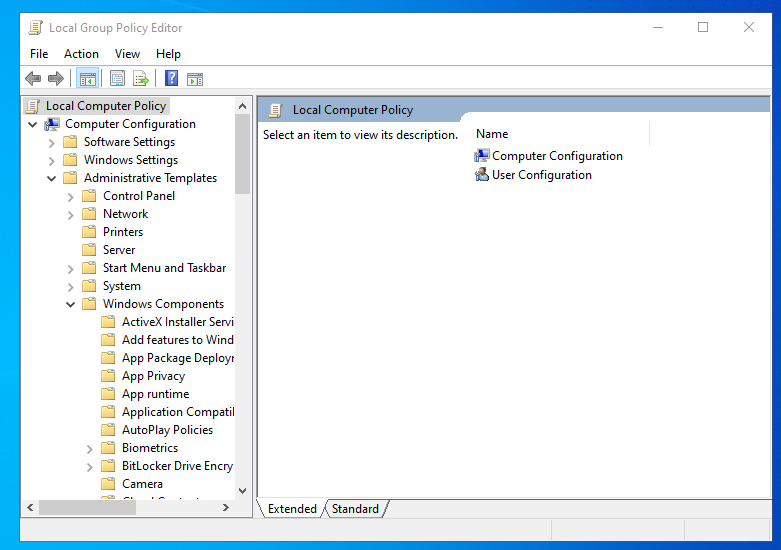
1. Check the time and date settings of your server, and correct them if necessary.



1. Updates are a crucial element of servers and systems. **Check for updates** and bring your system completely up to date.



1. In this lab environment it might become troublesome to wait for updates when we are in hurry. Therefore we will configure our server to only perform updates manually and not automatic. There is a tool called **sconfig** (we will use this one a bit later) that allows us to configure a lot, including Windows updates. Another option is the use of the **Local Group Policy Editor**. Make use of this editor to disable automatic updates (tip: gpedit.msc & gpupdate /force). Note: this is not recommended in a production environment!



And do not forget to disable it!

1. Restart the server and verify that your changes to the registry worked: is numlock enabled by default now?

Yes, it works great!

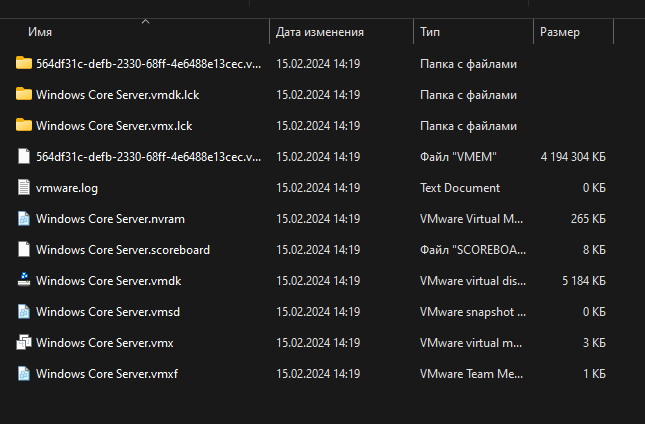
1. Shutdown the virtual machine, create a snapshot and offline backup.

## Windows Core Server

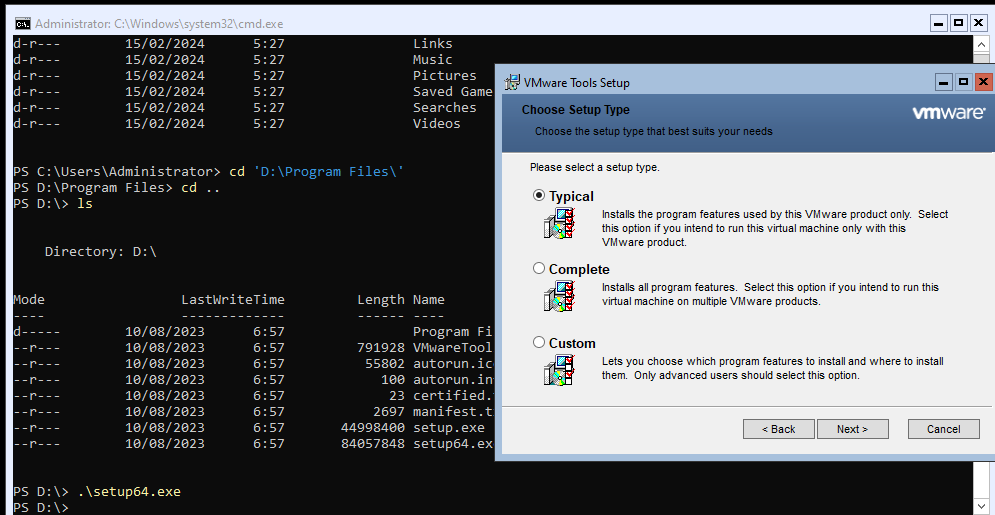
1. Perform the same steps as before to create a Windows Core Server. Name the server “**Windows Core Server**” and make sure to select the option “**Windows Server 2022 Datacenter**” (not the Desktop Experience).

The password is **Friday13th!**

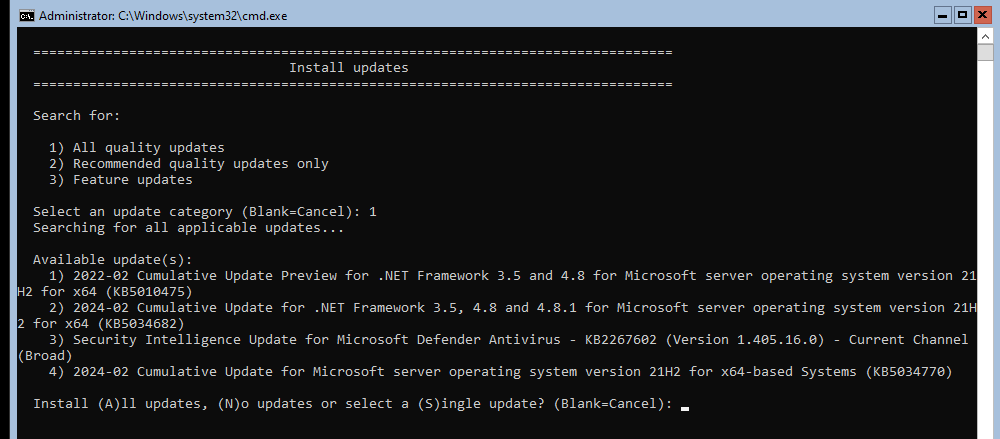
1. Browse to the location of your virtual machine configuration files (for the core server), and take a look at the vmdk-file. How big is it?



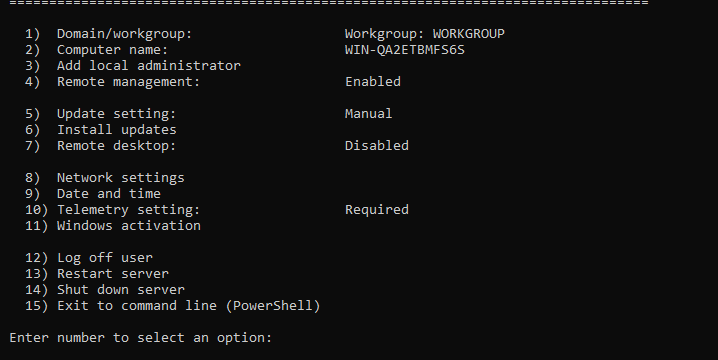
1. To install the VMware tools, exit to the command line and go to the D: location by typing “**D:**” in cmd (no cd or something else) and run the setup64.exe. Tip: The installation wizard may be hidden behind the terminal window. Move this window or use task manager (Tip: CTRL+SHIFT+ESC) to bring the installation wizard to the front.



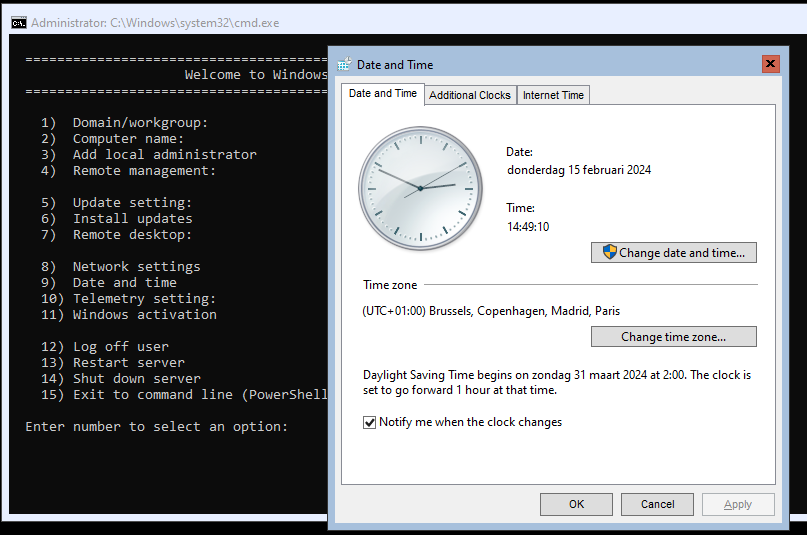
1. Run **sconfig** and perform a Windows Update and reboot if necessary.



1. In the sconfig menu find and change the option to turn off automatic Windows Updates.

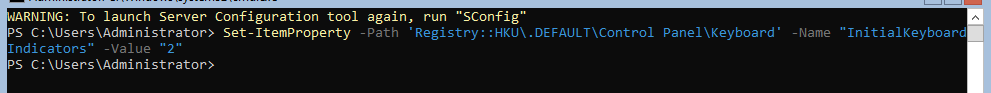


1. In sconfig, check that the time and date settings of your server are right, and correct them if necessary.



1. To enable numlock when logging in you can once again change the register value like before but this time we suggest using PowerShell.

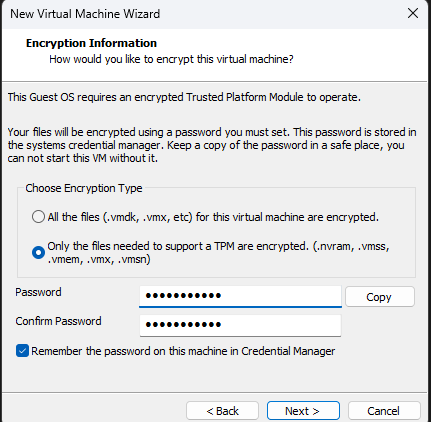
Set-ItemProperty -Path 'Registry::HKU\.DEFAULT\Control Panel\Keyboard' -Name "InitialKeyboardIndicators" -Value "2"



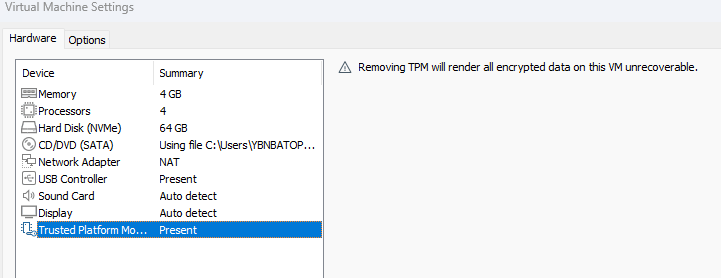
1. Finalize, by shutting down the virtual machine, creating a snapshot and making an offline backup.

## Windows 11 Client

1. Now we will create a new virtual machine for the **Windows 11 Client** (choose this as name). Because Windows 11 has different minimum hardware requirements than Windows server 2022 and even Windows 10 (see [Windows 11 requirements - What's new in Windows | Microsoft Docs](https://docs.microsoft.com/en-us/windows/whats-new/windows-11-requirements)), we’ll need to modify the VM configuration steps accordingly:
   1. Choose “Windows 10 and later x64” as operating system (a specific setting for Windows 11 is not yet available)
   2. Allocate 4 GB of memory space and 64 GB of disk space to the VM (not the default values!)
   3. After finishing the wizard, we need to add a virtual TPM module to the VM:
      1. Click Edit virtual machine settings à options à access control à Encrypt and set the password for the VM (suggestion: Friday13th!)

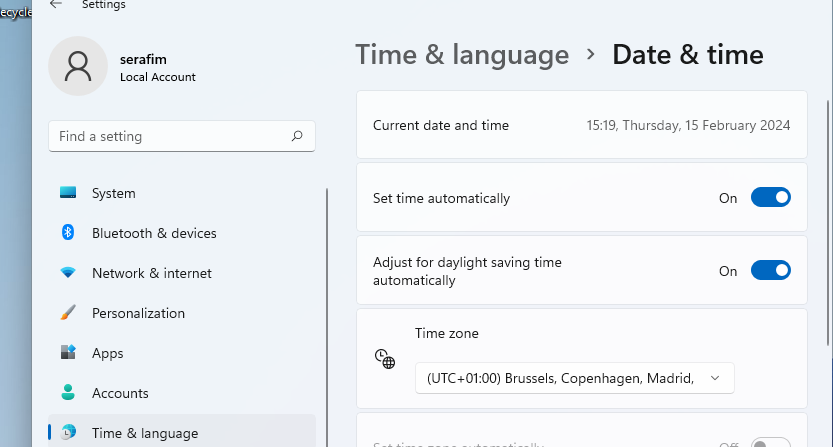


* + 1. Go back to the Hardware tab (Edit virtual machine settings à Hardware), click the “add” button and select the “Trusted Platform Module”.

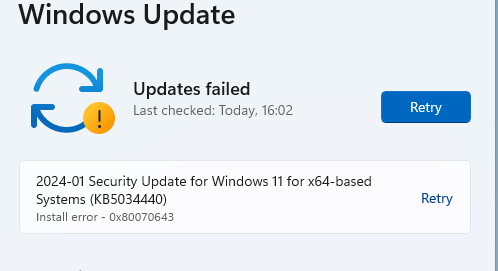
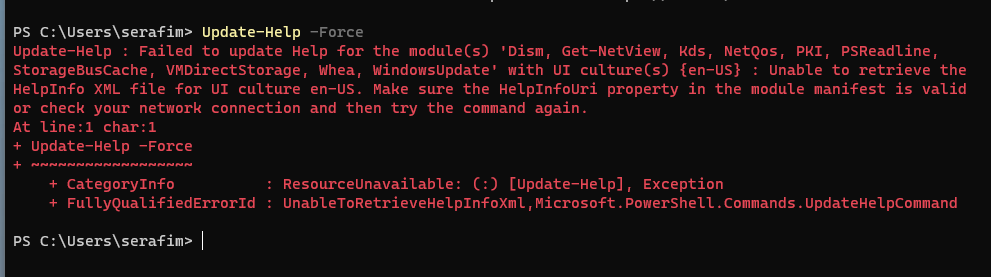


* 1. Use the Windows 11 ISO-file as install disk

1. Now start the VM to install Windows 11.
   1. Again, use “**Dutch (Belgium)**” as time and currency format, and a keyboard layout as you see fit.
   2. Choose to install the “Windows 11 Enterprise” version
   3. When asked for an account, don’t sign in with a Microsoft account, but click **“Sign-in options”** and choose **“Domain join instead”**
   4. Choose your **firstname (lowercase)** as username and remove any ¨,^,’… in your name. You are free to choose a password yourself but we recommend once again Friday13th!
   5. Answer 3 security questions
   6. Answer all remaining questions with “no”
2. Once again install the VMware tools.
3. Check the VM’s time and date, and correct these if necessary

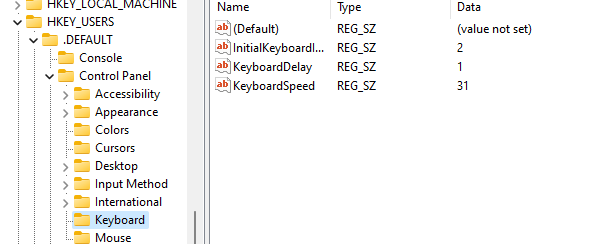
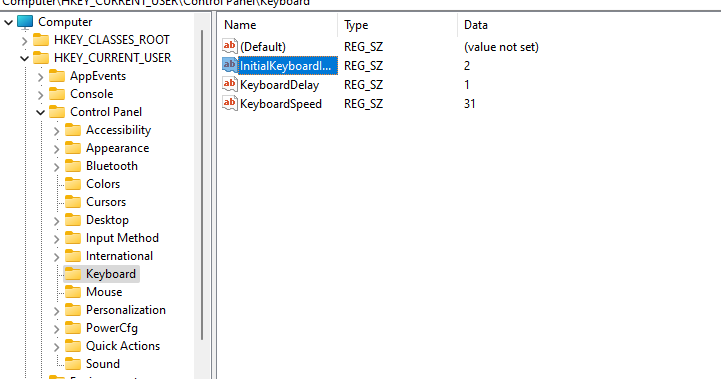


1. Once again perform Windows Updates and perform an “**Update-Help -Force**” in a PowerShell window as administrator. It might give an error for certain cmdlets but it performed an update nonetheless.

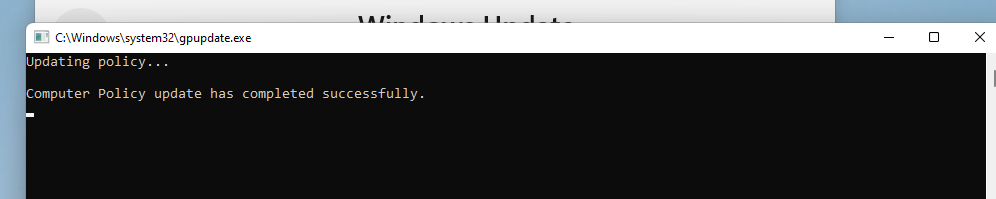
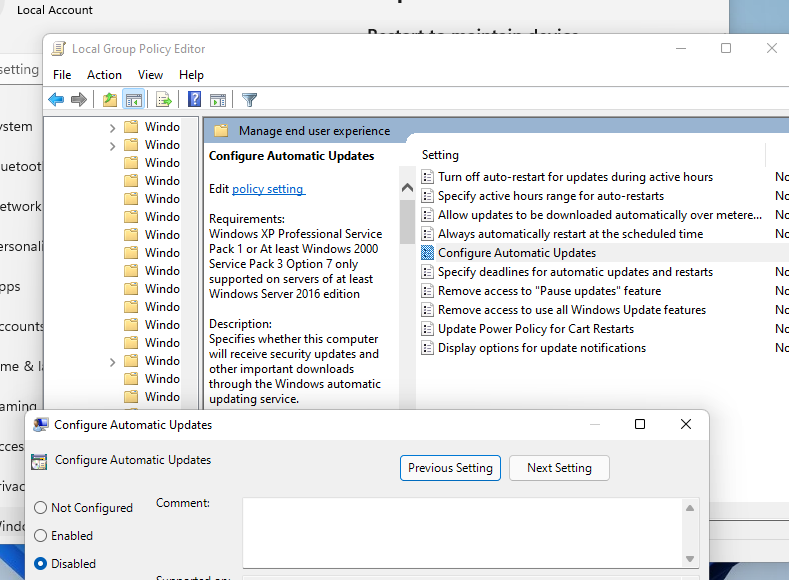


One of my updates keeps failing but I suppose it works fine

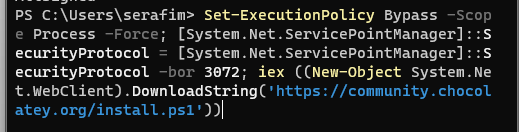
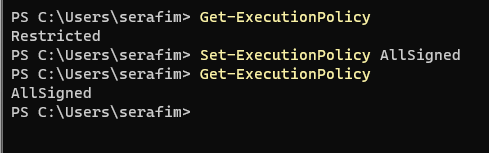
1. If necessary, change the numlock settings in the registry (refer to question 7).



1. Use **gpedit.msc** once more to disable automatic Windows updates and perform a **gpupdate /force**



1. To further prepare this client as a working machine we will install some extra tools. We could do this manually but (hopefully) you already tasted the functionality of package managers (on Linux for example) or scripts. Microsoft (finally!) worked on a package manager called “**winget”.** The tool is still in its infancy so we will also use another package manager that is community driven called **chocolatey.** Go to <https://chocolatey.org/> and follow the installation instructions. Read the instructions carefully!



* 1. What is this “**execution policy**” you see in the installation instructions of chocolatey.?

Set-ExecutionPolicy Bypass –Scope Process

* 1. What is the **default** execution policy?

Restricted, as seen in one of the screenshots above

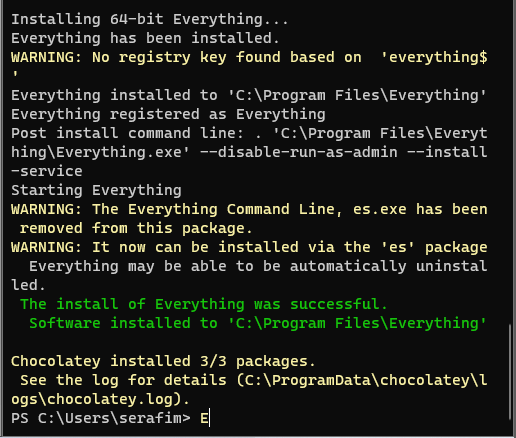
* 1. What are the **possible options** and how do they **differ** from each other?

Restricted, AllSigned, Bypass, Undefined, Default (Restricted for Windows, and RemoteSigned for Windows Servers), seems to be it.

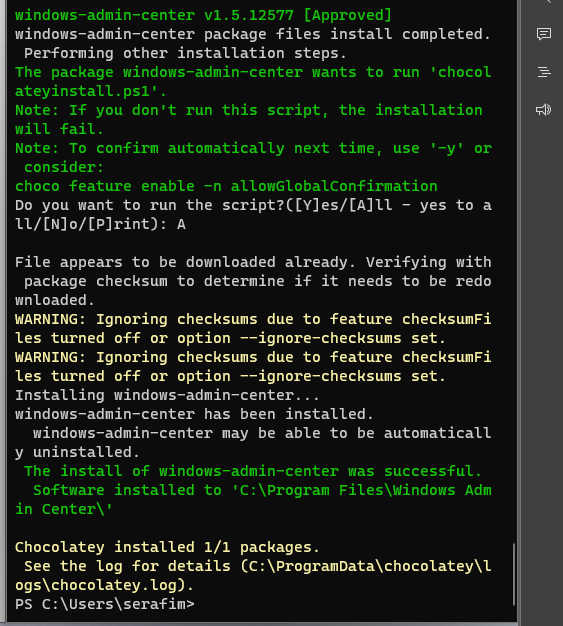
An execution policy is part of the PowerShell security strategy. Execution policies determine whether you can load configuration files, such as your PowerShell profile, or run scripts. And, whether scripts must be digitally signed before they are run. (Microsoft Documentation)

Overall this seems to be a term like with python-pip, that it makes sure you are allowed to run the script itself, and it depends whether it was signed or not, or from what authority it comes from.

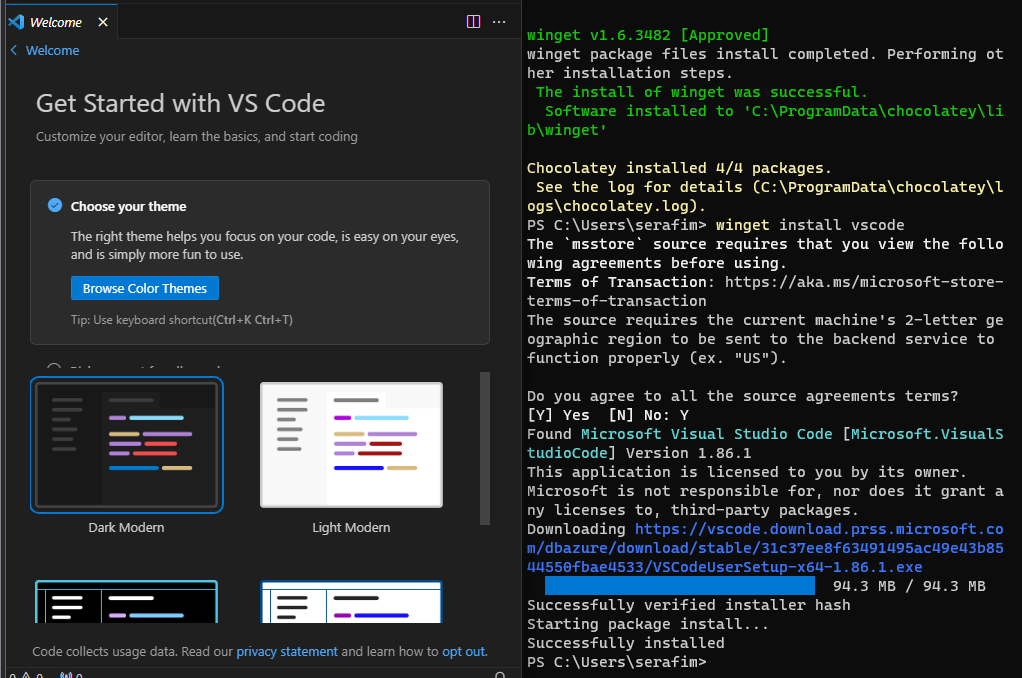
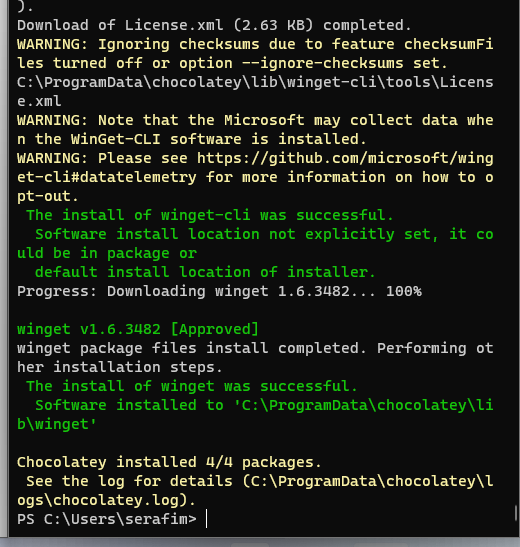
1. Once chocolatey is installed. Use the tool to **install** the following packages:
   1. Everything



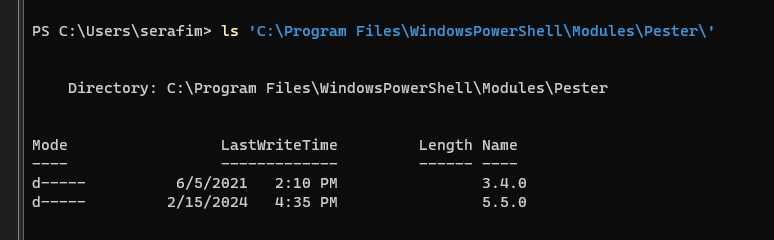
* 1. windows-admin-center



1. Let’s use the new **winget** tool from Microsoft as well. Use the winget command (install it first from the Microsoft store under the “**app installer bundle”**) to search and install visual studio code.



1. We will also use of the Pester framework. By default version 3.4 is already present but that’s not the latest version. Verify the contents of the following location: 'C:\Program Files\WindowsPowerShell\Modules\Pester\'. Use the following PowerShell one-liner to install the latest version of Pester: **Install-Module -Name Pester -Force -SkipPublisherCheck** Afterwards both 3.4 and 5.4 should be present. You can also verify with **Get-Module Pester -ListAvailable**



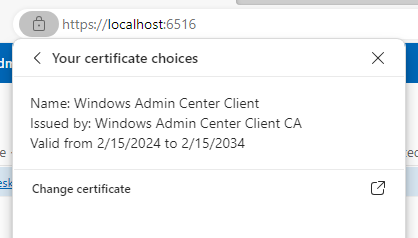
1. Pin vscode and Windows Terminal to your taskbar. The terminal is a quality of life tool for any Windows user that performs tasks on the commandline. Visual Studio Code is a free editor and Pester is a PowerShell test and mock framework we will use.

1. Windows Admin Center is a modern system configuration tool by Microsoft, that can be used to (remotely) manage client and server systems, both on-premises and in the (Azure) cloud. When it is started for the first time, some additional configuration steps are needed:
   1. Start Windows Admin Center in the start menu. What port is it using?

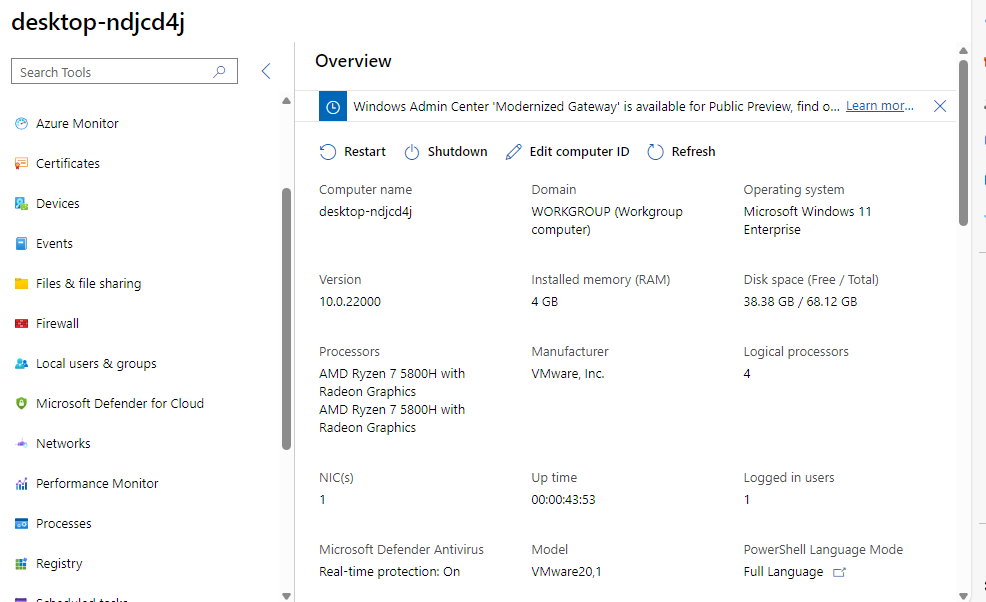
6516

* 1. Agree to (always) use the Edge Browser to open the configuration website
  2. Check the properties of the “Windows Admin Center Client” credentials. How long will this certificate be valid?

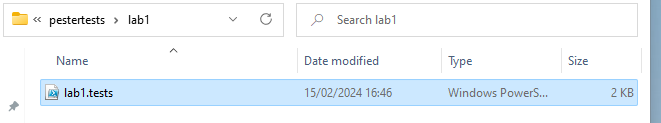
Till 15.02.2034



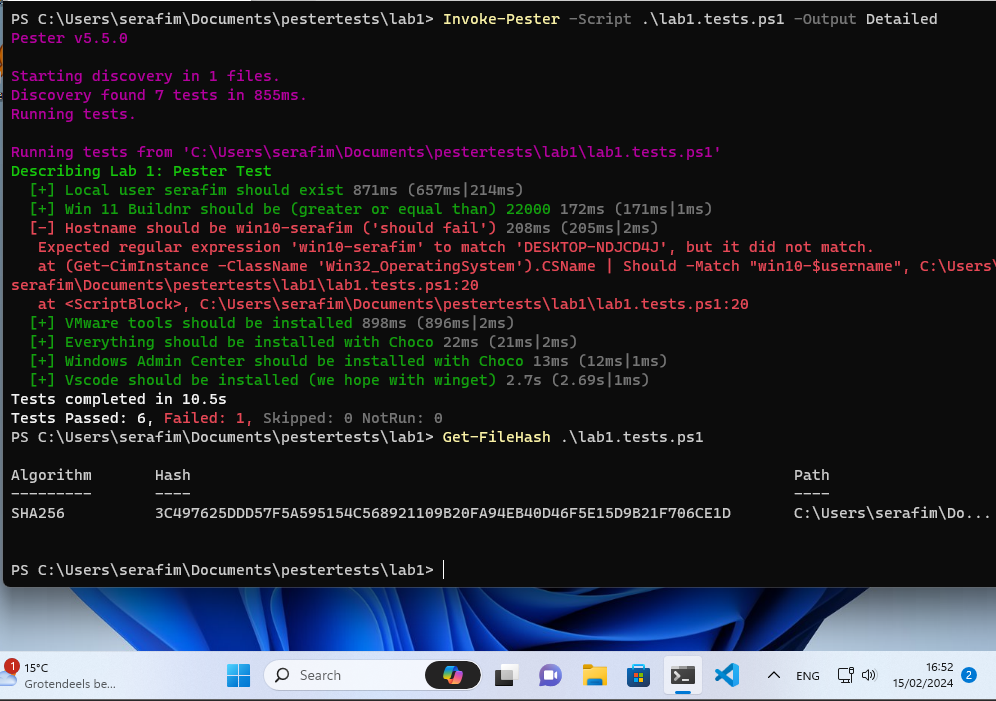
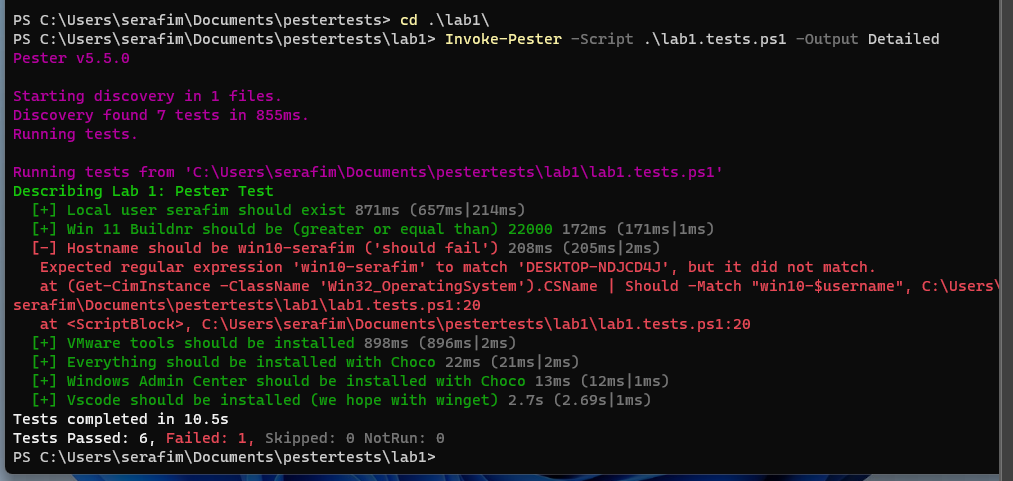
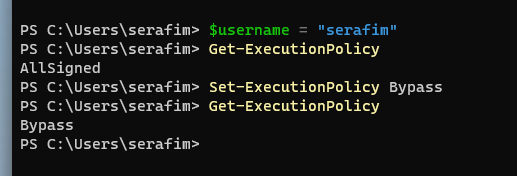
* 1. Select to use the “Windows Admin Center Client” credentials and press OK.
  2. Windows Admin Center will now try to connect to the local machine, but might fail due to a required service that is not yet running. Restart the Windows 11 machine if necessary (this will also start the required service), and start Windows Admin Center again. Allow automatic updates for the plugins.
  3. Connect to the localhost machine, and explore the administration interface to see what information you can find/modify.



1. On Leho you will find a pester-test file for this lab. In the documents folder of your user create a folder “**pestertests”**. In the folder create another directory called “**lab1**”. Put the pester-test file (from Leho) in this folder.



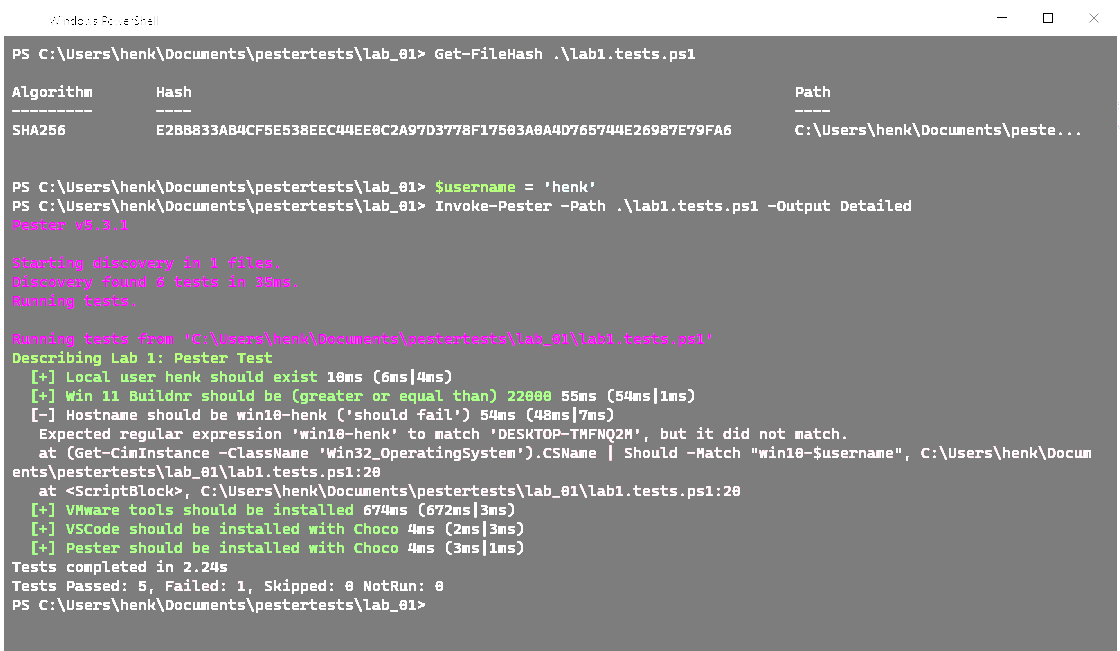
1. Open Windows Terminal and open a PowerShell window. Navigate to the lab1 directory and issue the cmdlet **“Invoke-Pester -Script lab1.tests.ps1”.** You should receive an error. Fix this and try again.



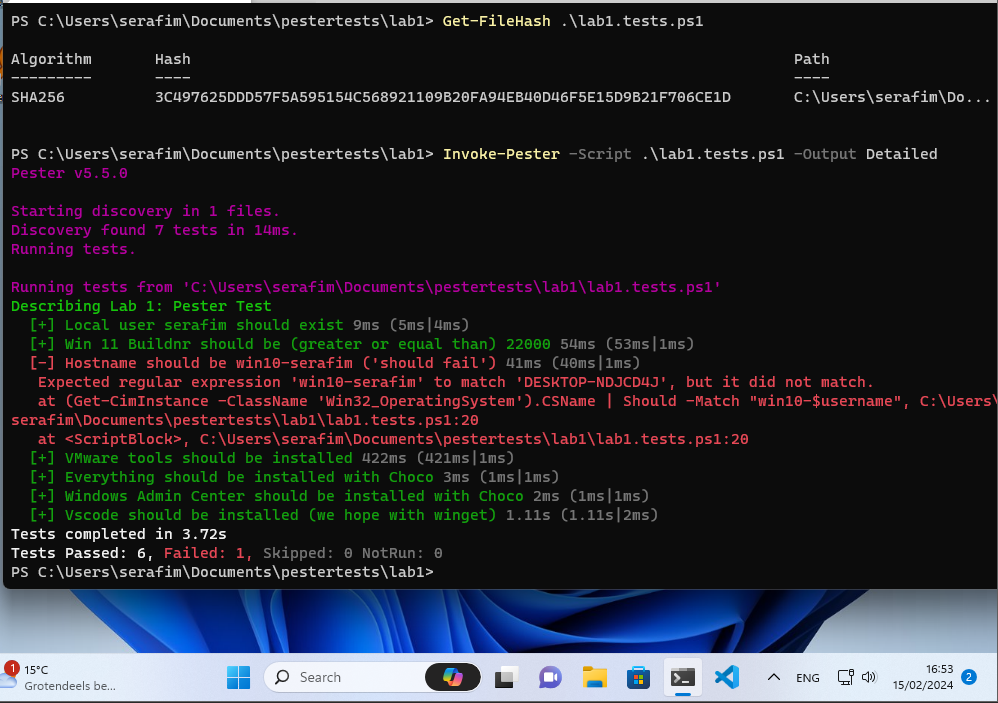
1. By default we only get a list of failed tests. Let’s fix this by running the Invoke-Pester with the parameter “-output detailed”. If everything goes right you should see a few **passing tests** and **one** that **fails. Create a screenshot as shown below (with the file-hash and the passing/failing tests).** (We suggest to type and not copy paste this one, for future labs we will use an alternative path but for now follow these instructions).

$username = ‘<your firstname>’

Invoke-Pester -Script .\lab1.tests.ps1 -Output Detailed



This is mine, hash first and then the tests



1. Finalize, by shutting down the virtual machine, creating a snapshot and making an offline backup.