Lab 7: Remote Access

Windows Server Labs

SERAFIM CIOBANU

Academic Year: 2022-2023

Table of Contents

[Introduction 3](#_Toc71177805)

[Learning Goals 4](#_Toc71177806)

[Knowledge (what you need to know) 4](#_Toc71177807)

[Abilities (what you need to be able to do) 4](#_Toc71177808)

[Requirements 4](#_Toc71177809)

[Remote Desktop Services 5](#_Toc71177810)

[Extensions - Optional Assignments 9](#_Toc71177811)

## Introduction

In previous labs we introduced multiple ways to interact with and manage Windows machines. Server manager, PowerShell Sessions and WAC are some examples. Some other ways such as SSH for Windows, Windows Remote Management and VNC will be introduced in other courses but feel free to explore these options to your own liking.

In this lab, we’ll explore one more method of remote access that can be used to manage both Windows servers and Windows clients, called the “Remote Desktop Protocol” (RDP). Additionally, we’ll also summarise some of the other access methods that were seen before, and have a look at how they integrate with each other and remote desktop.

## Learning Goals

# Knowledge (what you need to know)

* What is remote desktop and how do you start it.

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

* Manage machines with remote desktop
* Configure remote desktop

## Requirements

Your Win11, CORE and GUI machines are part of the same Active Directory domain, and can access each other.

## Remote Desktop Services

1. The Remote Desktop Protocol (RDP) is a network protocol developed by Microsoft to “securely” connect to an external desktop of a GUI server. However, even though RDP is called secure, remote desktop connections are not enabled by default. Check this in the following 2 ways (make sure that at least your GUI and Win11 machines are fully booted up before doing this!):
   1. Sign in to your Windows 11 VM with your personal domain user account and try to connect to your GUI server using the tool: “mstsc” (win + r). Does this work? After some time, a popup window will provide a number of possible reasons/explanations. What is - according to you – the real/main reason it fails?

In my case, it worked first time, probably because I had the right firewall rules enabled!

A screenshot of a computer

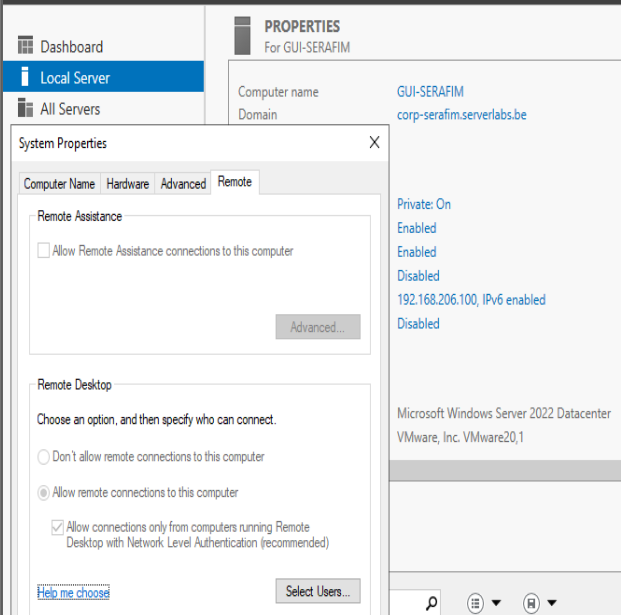
Description automatically generated

* 1. Sign in directly on your GUI server with the built-in (domain) Administrator account and check the **properties** of the **“local server”** in Server Manager. Take a look at the line “Remote Desktop”. It is indeed disabled by default. Change this default setting to ensure that your GUI server is accessible via RDP. Make sure that you add **user authentication** as an additional security measure. Use NLA = Network Level Authentication. Paste a screenshot below where you changed this setting.

Server Manager > Properties

A screenshot of a computer

Description automatically generated



A screenshot of a computer

Description automatically generatedAlso can be done via WAC easily, but in my case they are greyed out for some reason for the administration.

Me and Mr. Kenzo tried to solve it manually, but it does not help, hence no one can manage the remote desktop settings. It has to do with some rules that allow or restrict users to

1. Based on the default configuration of your now-enabled RDP, which users can create an RDP connection to the GUI server? TIP: use the “select users…” button. Do not add any extra users.

A screenshot of a computer

Description automatically generated

Any administrator and only the users that are going to be specified (hence in my case everything worked first try)

1. Test the custom setting by creating an RDP connection to your GUI server from the Windows 11 VM. Use the options below (via the “show options” button).
   1. Computer: The **name** of your GUI server
   2. User name: Your **personal** domain user account
   3. The local disk drive (i.e. the disk of your win11 VM) should be accessible in the RDP session.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Test this last setting, after you’ve created an RDP connection. Open Windows file explorer inside the RDP session and go to “This PC”. Paste below a screenshot that shows that the Win11 drive is available in the RDP session.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Yes, you can access it now, after specifying one rule

1. Go back to the GUI server VM’s interface in VMware workstation (i.e. not the RDP session). Was the Administrator logged out automatically?

No, he was not. (because it is not my administrator, but local administrator)

1. Now we’ll find out how many users can simultaneously create an RDP connection to the GUI server. **From your host** (laptop), create an RDP connection to your GUI server using the following options:
   1. Computer: the IP of your GUI server
   2. User name: your personal domain user account

Please note that MAC users will need to download and install the Microsoft Remote Desktop client tool. Why do you think you get a warning about the certificate that is used by RDP?

A screenshot of a computer

Description automatically generated

1. Is the RDP from the win11 VM still up, are you still logged in on the GUI? Finally, switch back to an RDP session from your win11 VM.

A screenshot of a computer

Description automatically generated

Yes, you are getting kicked out

1. On your GUI server VM’s interface (directly, not the RDP session), use Task manager to:
   1. Verify that 2 users are indeed logged in to your GUI server (1 via console, 1 via RDP from the win11 VM). Paste below a screenshot of the complete task manager window, showing all logged in users.

A screenshot of a computer

Description automatically generated

* 1. Do the same thing with PowerShell using the get-ciminstance cmdlet with the classname win32\_computersystem and selecting the username(s). Do you see 2 users?

Get-CimInstance -ClassName Win32\_ComputerSystem |

Select-Object -ExpandProperty UserName |

Where-Object { $\_ -like "\*\\*" }

A computer screen shot of a blue screen

Description automatically generated

It shows only one user, being the local administrator, who is logged in locally, and not remotely.

* 1. Send a message from the Administrator to your personal domain user account (check in the RDP session if you have succeeded)

Task Manager > Users > Right-Click on user > Send Message

A screenshot of a computer

Description automatically generated

A screenshot of a computer error

Description automatically generated

* 1. From the Administrator’s session, disconnect your personal domain user account and paste a screenshot of the message you get on your win11 VM.

Task Manager > Users > Right-Click on user > Disconnect

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

1. Start the RDP session once more from your win11. On your GUI server, use the legacy Windows command **query user** and **query session** to get information about users and sessions on your GUI server. Disc onnect the RDP session and run the commands again and note the difference. Paste below a screenshot of both commands along with their output.

Before

A computer screen shot of a blue screen

Description automatically generated

After

A screenshot of a computer screen

Description automatically generated

The query user now only shows local administrator.

The query session shows the local administrator session too, and says that the other one is disconnected.

1. Through task manager, you can **disconnect** users from the server, but also permanently log out users. The legacy Windows command logoff also has an option (use /? for more information) with which the Administrator can completely log out your personal domain user account. Try this and verify with the commands of the previous question(s).

A blue screen with white text

Description automatically generated

logoff <sessionID>

logoff 3

A computer screen shot of a blue screen

Description automatically generated

A screenshot of a computer

Description automatically generated

Now it was surely logged out.

1. From your host (laptop), create an RDP connection to your GUI server using the following options:
   1. Computer: the IP of your GUI server
   2. User name: your personal domain user account
2. Create a new domain administrator account on your GUI server named Administrator2 by copying the built-in Administrator account. Because we copy, the new user will be immediately be a member of the same groups like the first Administrator.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

name: administrator2

password: Friday13th!

From your host, now try to make a second (simultaneous) RDP connection to your GUI server, using the following options:

* 1. Computer: the IP of your GUI
  2. User name: the new administrator2 account

A new window will appear on your screen (paste a screenshot below) and disconnect your personal user account. After issuing the disconnection go to your RDP window of your personal user account. You should be prompted the question to either agree or disagree. If you don’t do anything you will be automatically disconnected after a few seconds, how many seconds?

A screenshot of a computer

Description automatically generated

A blue screen with white text

Description automatically generated

It will automatically disconnect you after 30 seconds.

## Mixing it up: WAC, Remote Desktop and (Remote) Powershell

As we’ve seen before, Windows Admin Center (WAC) is a new, browser-based remote management interface for Windows machines. To allow for more flexibility in administering these machines, Microsoft has integrated a number of other tools, like RDP and Powershell, into WAC. In the following section, we’ll explore some of the possibilities this provides. Now, we’ll remotely manage the CORE server, so make sure that all three virtual machines (GUI, CORE and Win11) are booted up (the GUI server is always needed to provide DNS services).

1. Log in to the Win11 VM with your domain administrator account, start Windows Admin Center and connect to the CORE server. Go to the “Remote Desktop” tool page, and try to remotely log in (allow to connect with the provided certificate). Does this work? Why (not)?

In my case it works, as probably I have allowed the remote desktop access to other users, and also set up the correct policies.

A computer screen shot of a computer

Description automatically generated

1. Like with the GUI server, we’ll first need to change the server’s configurations. Fortunately, this time we don’t need to go to the server itself to change this setting. Go to the “Settings” tool page in WAC (bottom left) and correct the required setting. Do you notice something missing, compared to the configuration window in question 1.b.? WAC is still under development, and from time to time, you’ll find some omissions like this.

A screenshot of a computer

Description automatically generated

I believe that I need to go here, as this is also the place where you can set up the RDP. And moreover, it is said that I do not have to go to the server itself, hence I think I did everything right (especially considering everything works for me). And I see no issues with the interface right now, hence I believe you can still find bugs and etc.

1. Now go back to the Remote Desktop tool page, and try to log in to the CORE server. Does sconfig show that remote desktop connections are possible? Use the legacy commands to again check the currently active users and sessions on the CORE server.

A screenshot of a computer

Description automatically generated

Yes, it seems like enabled.

A screenshot of a computer

Description automatically generated

I was logged in as my personal administrator, hence I was logged off from the CORE server, and hence have only one session.

1. Now let’s look at the integrated remote powershell feature in WAC. Go to the “Powershell” tool page, and notice that a remote powershell environment is set up automatically. By pressing the “up” arrow key, you can see some of the commands that were used to do this. Was a remote desktop session initiated for this? (check this with the legacy “query” command). As what user are you working on the core server?

A computer screen shot of a program

Description automatically generated

It does not look like I am actually using a user, but it is more like some kind of remote service that I am representing, according to “query session”

1. In its back-end services, WAC actually uses powershell commands to configure and retrieve settings of the remote host. Microsoft even decided to include some samples of the code they used for this in the WAC interface. For each of the tool windows, you can find this sample code by clicking the “>\_” button at the top right of the interface. Search for the code that is used in the Overview window to retrieve properties of the server’s Network Interface Cards, and test if you can run this code yourself in the powershell window for the CORE server. Take a screenshot of the results and paste it below.

Get-CimInstance -Namespace root/cimv2 -ClassName Win32\_NetworkAdapter

A computer screen shot of a computer

Description automatically generated

Seems to do it.

## Extensions - Optional Assignments

1. From your host, you should be able to configure all 3 machines to have remote access using the Enter-PSSession and Invoke-Command. Check if this is already possible for your machines and if not make sure it works.

Search a new PowerShell cmdlet to create PS-sessions instead of entering one. Figure out a way to create 3 sessions to each of your virtual machines (the gui, the core and the win11). Using a PowerShell oneliner retrieve the IP addresses of all machines using the previously created sessions.

winrm set winrm/config/client '@{TrustedHosts="192.168.206.100"}'

winrm set winrm/config/client '@{TrustedHosts="192.168.206.101"}'

winrm set winrm/config/client '@{TrustedHosts="192.168.206.102"}'

This is used to actually add the computers to connect to, into the trusted hosts, and hence we could connect. But you need to do it one by one.

And then you will need to specify even the Credentials for the computer you want to connect to, along with the IP address.

Enter-PSSession -ComputerName 192.168.206.100 -Credential CORP-SERAFIM\ciobanus

To create a new session, you can then use this comman

New-PSSession -ComputerName 192.168.206.100 -Credential CORP-SERAFIM\ciobanus

So I added the IP addresses to the trusted hosts one by one, and also been opening sessions to them one by one, after adding to trusted hosts.

Now I can see the sessions

Get-PSSession

A screenshot of a computer screen

Description automatically generated

And to get the IP Addresses (IPv6 + v4):

Get-PSSession | ForEach-Object {

Invoke-Command -Session $\_ -ScriptBlock { [System.Net.Dns]::GetHostAddresses($env:ComputerName).IPAddressToString }

}

A screen shot of a computer

Description automatically generated