Lab Goals:

This lab will be a hands-on lab to give you first-hand experience with accessing Windows Server remotely. We will be using several access methods to accomplish different responsibilities.

We will be working on the following SLOs during this lab:

- Students will be able to enable, and log into a remote Windows machine via Remote Desktop for Administration.
- Students will be able to use Powershell to connect to a remote Windows machine for administration.
- Students will be able to utilize the PsExec utility to launch processes on a remote system.

Grading: You will be required to turn in answers for the questions in this lab, and your code written.

Resources: https://technet.microsoft.com/en-us/sysinternals/default

Lab Prerequisites:

This lab builds on previous CS338 labs. You will need to utilize the VMWare vSphere web client to connect to your assigned hypervisor. You will be using your GUI VM server.

You will need to obtain the following items from the instructor during the lab:

- None.

Please connect your vSphere web client to your assigned hypervisor at this time. Log in using your CSAD\<username> credentials.

Lab Notations:

Please note the the following notations will be used in the lab.

< > - anything in a < > bracket is intended for you to fill in with the appropriate item. For example, <username> would be filled in with your username. If my username was abc1, then <username> = abc1 .

Lab Exercise:

1. Remote Desktop for Administration - GUI

The first way we will be accessing Windows server is via Remote Desktop for Administration. This will allow us to have a full GUI-based connection to the Windows server for full Administration like you are at the console. We will be enabling this first via Server Manager, and then via the sconfig console for our core server next section. As a side-note, Remote Desktop is frequently shorted to "RDP", and you will see it used frequently as such during this lab.

Log in locally to both of your Windows servers using your CSAD\<username> login.

a. On your "338-<username>-g" server, open Server Manager.
b. In the Local Server area, click on the "Remote Desktop" option.
c. "Allow remote connections for this computer" and "only from computers running Network Level Authentication (NLA)". Click "OK" to close out the wizard.

Question 1: What firewall ports are opened up when you enable RDP? (Google it)

Once you have configured this, we want to make a Remote Desktop connection to the machine from our syslab machine. On your syslab machine, do the following:

d. Start -> Accessories -> Remote Desktop Connection.
    i. Computer: 338-<username>-g.csad.cs.calvin.edu
e. Connect!

Assuming everything goes alright, you will connect up to your Windows server using Remote Desktop and have full console access to the machine. Verify that this is working correctly, pull up your Firefox browser to test out your connection.

Question 2: What happens to your console in VMWare when you connect via RDP?

f. Next, we are going to close out our RDP session. From the Start Menu, select the "People" icon on the far left, and choose "Sign out".

Using RDP, we can do some fun little tricks. One of the nice things about RDP is that you can map your local computer's drives to the remote server for ease of copying files, etc. We will be enabling this functionality this time when we connect to our GUI server.

g. Open up Remote Desktop Connection.
    i. Show Options

1. General:
    a. Computer: 338-<username>-g.csad.cs.calvin.edu
    b. User name: CSAD\<username>
2. Local Resources
    a. Local devices and resources
        i. More…
            1. Drives - check all
3. General
    a. Connection settings
        i. Save As…
            1. Save your RDP connection as "338-<username>-g.rdp"

ii. Connect!

Turn in 1: Turn in your 338-<username>-g.rdp file with this lab.

If everything connects correctly, you should now have some extra entries under your "My Computer".

h. Open up "My Computer"
    i. Under "This PC", check for additional drives that look something like "C on SYSLAB<##>".

i. Use "My Computer" to copy a file from your local C drive to the remote C drive.
    i. Local file: C:\Windows\setupact.log
    ii. Remote server file: C:\installers\setupact.log

Question 3: What other local resources can Remote Desktop automatically redirect to our server?

Another "feature" of Remote Desktop that we will look at is the ability to leave our session running on the remote server and resuming it where we left off.

j. Close your Remote Desktop connection using the "X" at the top of the program. Make sure your File Explorer window is still up. Do NOT sign out of the server.
k. Reconnect to your Server using Remote Desktop Connection.

Question 4: How does RDP behave when you sign out of the server vs. just closing your Remote Desktop program?

One last thing we will do will show the limitations of Remote Desktop for Administration. You will need a two partners for this section. Group up with a few individuals (or ask your professor for assistance if needed.) You will only have to do this for one individual's server.

Make sure you are logged into your server using your CSAD\<username> credentials over RDP before continuing. Do NOT close your connection.

l. Have your two partners simultaneously establish connections to YOUR GUI server, logging in with their CSAD\<username> credentials.

Question 5: What happens? How many people are allowed simultaneously via Remote Desktop for Administration? (Google it)

2. Remote Desktop for Administration - Core

Open up the console for your core server via VMWare. Log into your core server using your CSAD\<username> account.

a. At the DOS prompt, execute “sconfig”
    i. Select “Remote Desktop”
        1. Enable Remote Desktop
            a. Allow clients only running Network Level Authentication
b. Logout of your core server using the “logoff” command.

We will now be making a connection to our core machine.

c. Use Remote Desktop Connection to establish a Remote Desktop connection to your 338-<username>-c server.

Verify that you got into your server correctly. If you got in, please continue on. Otherwise, find someone to help you before continuing.

d. Save off your RDP connection information as “<username>-2016-core.rdp”.

Turn in 2: Turn in your 338-<username>-c.rdp file with this lab.

e. Make sure you are logged off (using the “logoff” command) and have all remote sessions closed out at the end of this section.

3. Powershell Remote Connections

Resources:
http://www.howtogeek.com/117192/how-to-run-powershell-commands-on-remote-computers/, https://msdn.microsoft.com/en-us/powershell/reference/5.1/microsoft.powershell.core/enable-psremoting

We will be using Powershell to connect to our servers this time around. Perform this operation on BOTH of your GUI and core servers.

a. Open Powershell as an Administrator.
b. Perform the following PS commands:
    i. Enable-PSRemoting -Force
    ii. Set-NetFirewallRule -Name “WINRM-HTTP-In-TCP” -RemoteAddress 153.106.0.0/16

Once you have performed this command on your servers, we will try to make a connection via Powershell from our Syslab machines. We can run individual commands using the “Invoke-Command” cmdlet.

On your syslab machine,

c. Open Powershell

First, we will run just a single command on the remote server using the Invoke-Command cmdlet:

d. Run the PS command: Invoke-Command -ComputerName 338-<username>-g -ScriptBlock { ipconfig /all } -Credential CSAD\<username>

Question 6: What is the MAC address of your remote GUI server’s ethernet connection?

While useful (especially for scripting) to perform a single command, for Remote Administration you will sometimes want a persistent connection via Powershell to your remote server.

e. Run the PS command: Enter-PSSession -ComputerName 338-<username>-g -Credential CSAD\<username>

Once you have made your connection, pull up some information on your remote machine. Verify that powershell has connected successfully and then run:

f. Get-ComputerInfo

Question 7: When was your server installed according to the registry?

Exit out of all your Powershell sessions.

4. PSTools for Remote Administration

PSTools is a set of utilities from the SysInternals suite. While originally developed independently from Microsoft, Microsoft has acquired and hosts the SysInternals suite. See more information at https://technet.microsoft.com/en-us/sysinternals/default .

Work on your Syslab machine for this section.

a. Download the PSTools utility from https://technet.microsoft.com/en-us/sysinternals/bb897553.aspx . Unzip the PSTools.zip file into C:\installers\PSTools\
b. Open up a command prompt (as your normal user, NOT as an Administrator)
c. In your DOS prompt, navigate to: C:\installers\PSTools\

We will be using the PSExec and PSlist commands for this lab.

d. Run the command: pslist

Question 8: What does the pslist command do?

Next, we will be using the remote capabilities of pstools to connect remotely to the the servers SMB Admin$ share, and execute commands.

e. Run the command: pslist \\338-<username>-c

This should give you a listing of all the processes running on the remote machine. We're going to take at some of the options that we can use with pslist.

f. Run the command: pslist --help

Question 9: What command line switches would allow us to specify an alternate username to connect remotely? What switch would allow us to see a process tree?

Next, we're going to use the "psexec" command to remotely execute a command at the DOS prompt.

g. Run the command: psexec \\338-<username>-c "systeminfo"

Question 10: What is the output of the systeminfo command? How many hotfixes have been installed?

PSexec can be used this way to do several beneficial commands. One of the easiest use cases for PSexec is to perform a remote "gpupdate /force" on a machine. This can be used in scripts to quickly apply Group Policy to a list of machines.

PSexec can also be used to get an interactive command prompt on a remote machine.

h. Run the command: psexec \\338-<username>-c "cmd.exe"

    i. Wait for the PSexec command to connect.
    ii. Perform an “ipconfig /all” command to verify you are on the remote machine.

PSexec is a wonderful quick and easy command to do some “quick and dirty” administration of a remote machine. The PStools pack also contains programs to allow you to kill off remote processes, and locally (and remotely) manipulate processes on Windows servers.

i. Exit out of all your command prompts to finish this lab.

Lab Conclusion

Turn in your question answers and requested turn-in files to conclude this lab.