📬 Submit this assignment by **Monday, March 27th at 11:59pm PDT** using the *Submit* button 👉

## **Project 5: DIY Malware**

### **Overview**

In this unit's lab, we explored some test virus files and looked at how they're detected. In this project, you'll get a chance to peek behind the curtain and see how virus files are actually created.

You'll use [MSFVenom](https://docs.metasploit.com/docs/using-metasploit/basics/how-to-use-msfvenom.html) to create your own (relatively-harmless) virus files, then test them against virus scanners and VirusTotal to see if you can evade detection.

### **🎯 Goals**

By the end of this assignment you will be able to...

* Use MSFVenom to create a variety of virus files
* Check unknown files for embedded viruses with VirusTotal

### **📬 What You'll Turn In**

For this assignment, you'll be filling and submitting a copy of the [📄 **Project 5 Submission Template** (Google Doc)](https://docs.google.com/document/d/1CkodSp6kDhEo0tHSjTWxlt1xjTqj0TsjxF_4k__ueiA/copy)

* Before proceeding, we recommend you **open it up now** and read over the requirements in the document.
* It might be easier to "fill-as-you-go" than try to fill it all out after you complete the project.

In addition, you'll also be turning in **screenshots** of your kali terminal at various points in the task.

* To help you keep track of these, we'll put **📸 SCREENSHOT** tags throughout the instructions at the appropriate times.

Required Challenge

*(Detailed instructions on completing this project are provided below.)*

*To receive full credit, you must turn in..*

* **Screenshots showing creation of each virus file (Steps 1-3), including:**
  + Each command run
  + The output of each command run
* **Answers to each of the Project Questions in the submission doc**

You are **not** required to show:

* Package installation and setup
* Connecting to Kali (ssh or rdp)
* Virus file contents / execution

Close Section

Stretch Challenge

For bonus points, you may submit:

* **Screenshots showing checking all 3 virus files with the VirusTotal CLI**
  + Each command run
  + The output of each command run
* **Screenshots showing checking all 3 virus files with the VirusTotal Webiste**
  + Screenshot of VirusTotal results page

Close Section

## **Project Instructions**

🛑 **Stop! Make sure you read EVERYTHING in this box before proceeding.**

In this project, you will be **creating viruses**. Relatively harmless and easy-to-detect viruses, but still ones that in the wrong hands could cause harm to a computer. Like a scientist studying biological viruses, it's okay for us to do this, but ONLY in a safe laboratory environment.

**Virus Handling Safety Rule:**

**➡️ Do NOT take the viruses off your Kali VM ⬅️**

Really, that's all you have to do. If you remove them from your VM, even for your own *personal use* or to *prank a friend* or something else well-meaning, you could accidentally be 🚨 **committing a crime**.

The **only** exception to this rule is uploading the virus files **directly** to VirusTotal.com

Don't take that risk. Leave the files on the VM.

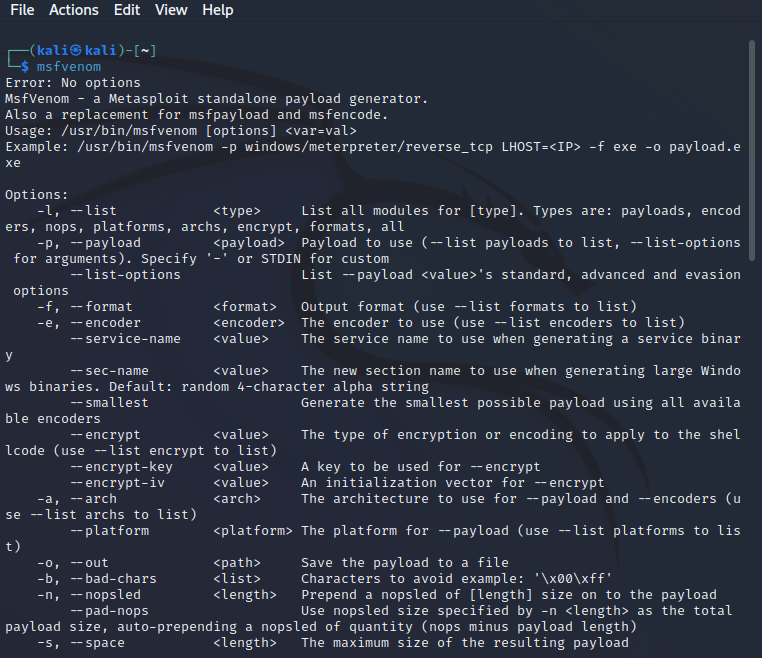
### **Required Challenge**

#### **Step 0: Start your Kali VM and check if MSFVenom is already installed**

In your Kali VM, open a terminal, and check to ensure MSFVenom is installed:

* Run the msfvenom command in Kali

Example Output



Close Section

#### **Step 1: Create a simple, single-payload virus**

We'll be creating viruses using msfvenom. It's syntax is as follows:

msfvenom -a ARCHITECTURE --platform PLATFORM -p PAYLOAD [ARGS] -f FORMAT -o OUTPUTFILE

Clear as mud? Let's go through those one at a time:

* **-a** is for the [computer architecture](https://en.wikipedia.org/wiki/Computer_architecture). This is something determined by the kind of CPU in the computer you're using. Most modern computers run on x64 or x86 architecture.
* **--platform** is for what [computing platform](https://en.wikipedia.org/wiki/Computing_platform) the virus should run on. For us, that means the OS/programming language, such as windows, osx, or python
* **-p** is the PAYLOAD. This is what the virus will actually *do*. MSFVenom offers hundreds built-in payloads, similar to metasploit modules.
* **-f** is the FORMAT of the output file. For example, .exe, .sh, .py, etc.
* **-o** is the output file -- the name of the virus you're creating!

All of that in mind, let's create the first virus. We'll use the windows/messagebox payload.

* Run the following command. Feel free to change the text.
  + msfvenom -a x86 --platform windows -p windows/messagebox TEXT="Virus Executed" -f exe -o messageVirus.exe
* Take a 📸 **SCREENSHOT** of the output for your submission doc
* Answer Project Question 1 in your submission doc

**🎯 Checkpoint 1**: You should have a single virus file, messageVirus.exe on your Kali machine.

#### **Step 2: Create a virus with multiple payloads**

Great, now let's see how multiple payloads can be bundled together into one virus.

First, you need to create the first payload. Let's use another messagebox, but this time we'll save it using the raw format:

* Run the following command. Feel free to change the text.
  + msfvenom -a x86 --platform windows -p windows/messagebox TEXT="Virus Executed" -f raw > messageBox

The command above will generate a file called messageBox.

Next, you can run another command with the -c flag to bundle in the payload we just created.

* Make the virus file with the following command:
  + msfvenom -c messageBox -a x86 --platform windows -p windows/speak\_pwned -f exe -o pwnedVirus.exe
* Take a 📸 **SCREENSHOT** of the output for your submission doc
* Answer Project Question 2 in your submission doc

**🎯 Checkpoint 2**: You should have a second virus file, pwnedVirus.exe on your Kali machine.

#### **Step 3: Create a virus with an encrypted payload**

Finally, you'll try to create an encrypted payload to see if that can slips past the anti-virus detection!

The following is the syntax we'll add to the command that will create an encrypted payload:

-e ENCODER -i NUM\_EXECUTIONS

To encrypt, we're going to wrap out payload in a python file, and encrypt it this time:

* msfvenom -a x86 --platform Windows -p windows/messagebox TEXT="Encrypted Virus" -e x86/shikata\_ga\_nai -i 3 -f python -o messageEncrypted

Then, using the encrypted file generated above, we'll create the virus file:

* msfvenom -c messageEncrypted -a x86 --platform windows -p windows/speak\_pwned -f exe -o pyVirus.exe
* Take a 📸 **SCREENSHOT** of the output for your submission doc
* Answer Project Question 3 in your submission doc

**🎯 Checkpoint 3**: You should have a third virus file, pyVirus.exe on your Kali machine.

### **Stretch Challenge**

#### **(Optional) Check in VirusTotal if the file are detected**

To complete this step, you'll have to use the VirusTotal CLI from lab 5.

* Using the VirusTotal CLI that you have configured in this week's lab, check whether the files that you have created are detected.
* Take a 📸 **SCREENSHOT** of the output for your submission doc
* Answer Stretch Question 1 in your submission doc

#### **(Optional) Upload these files to Virus Total**

Note: This is the **only** time the virus files should **ever** leave your Kali box!

Next, let's try uploading them to VirusTotal. Follow these steps to do so:

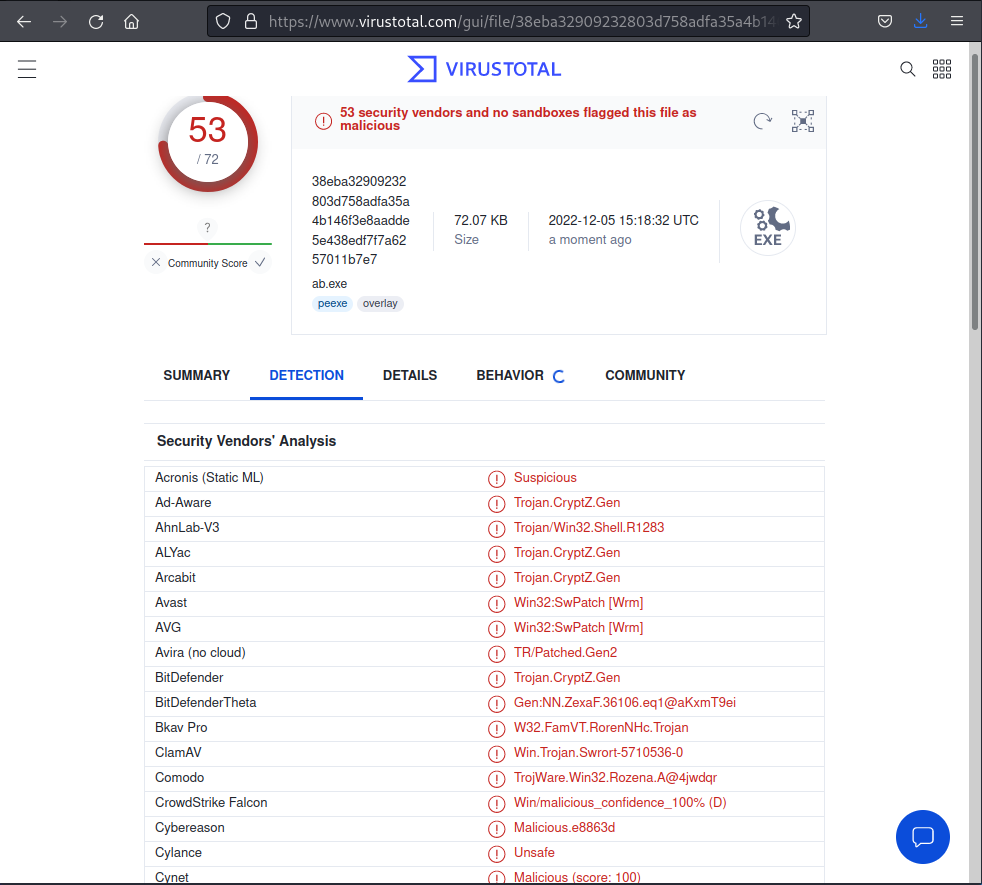
* Connect to your Kali VM via RDP
* Open your web browser (e.g., Falkon) and navigate to VirusTotal.com
* Upload each of the three virus files

🚨 Do **NOT** download them to your local machine!

Once VirusTotal has a chance to process them...

* Take a 📸 **SCREENSHOT** of the output for your submission doc
* Answer Stretch Question 2 in your submission doc

Example VirusTotal output



Close Section

Congratulations! Go to **"What You'll Turn In"** for instructions on submitting this project.

### **📬 Submitting Your Project**

* [📄 **Project 5 Submission Template** (Google Doc)](https://docs.google.com/document/d/1CkodSp6kDhEo0tHSjTWxlt1xjTqj0TsjxF_4k__ueiA/copy)

✔ Am I Ready to Submit?

Check if you're **ready to submit** with the following questions:

* Did you complete all of the **Required Challenges**?
* Did you copy and fill out the [**Project 5 Submission Template**](https://docs.google.com/document/d/1CkodSp6kDhEo0tHSjTWxlt1xjTqj0TsjxF_4k__ueiA/copy)?
  + It is important that you follow the same layout as the template so that we can easily access your work.
  + Be sure to check off each feature that is implemented in the **"Submission Checklist"** section
* Are any required images/GIFs correctly displaying in your document?
* Did you set your document to ***"Anyone with the link can Edit"***?

If you answered **yes** to **all** of these questions, you are ready to submit!

Look for the **"Submit"** button at the top of this page.

Close Section

📣 **Late Submissions**

* We highly encourage you submit your project in any state (even if it is not done) by the deadline: **Monday, March 27th at 11:59pm PDT**.
* You can continue to work on your project with our **48-hour extension** in which your project will be graded once more once the extension deadline has passed.
* Don't forget to **resubmit through the course portal** with your **updated document link**!