**CS361 Computer Security**

**Assignment 1**



Indian Institute of Information Technology, Guwahati

2101012 - Abhishek Kumar

2101081 - Harsh Rajput

Group - CS31

# **Introduction**

The purpose of this report is to provide an analysis of a program named 'V' and its potential impact on computer systems. 'V' is designed to execute specific operations, including the search, modification, and propagation of files with a .foo extension. The program primarily targets the user's documents folder, as well as any mounted USB drives connected to the system. Additionally, 'V' exhibits behaviour that involves copying itself to new computers when a specific directory within the documents folder is detected, when an infected USB is inserted to a new computer. The report aims to shed light on the actions performed by 'V' and observe the actions taken by antivirus as a response to this program 'V'.

# **Objectives**

## **Behavioural Analysis:** Examine the step-by-step actions performed by 'V' on execution, including the search for .foo files in the user's documents folder, infecting those files, and the search for mounted USB drives.

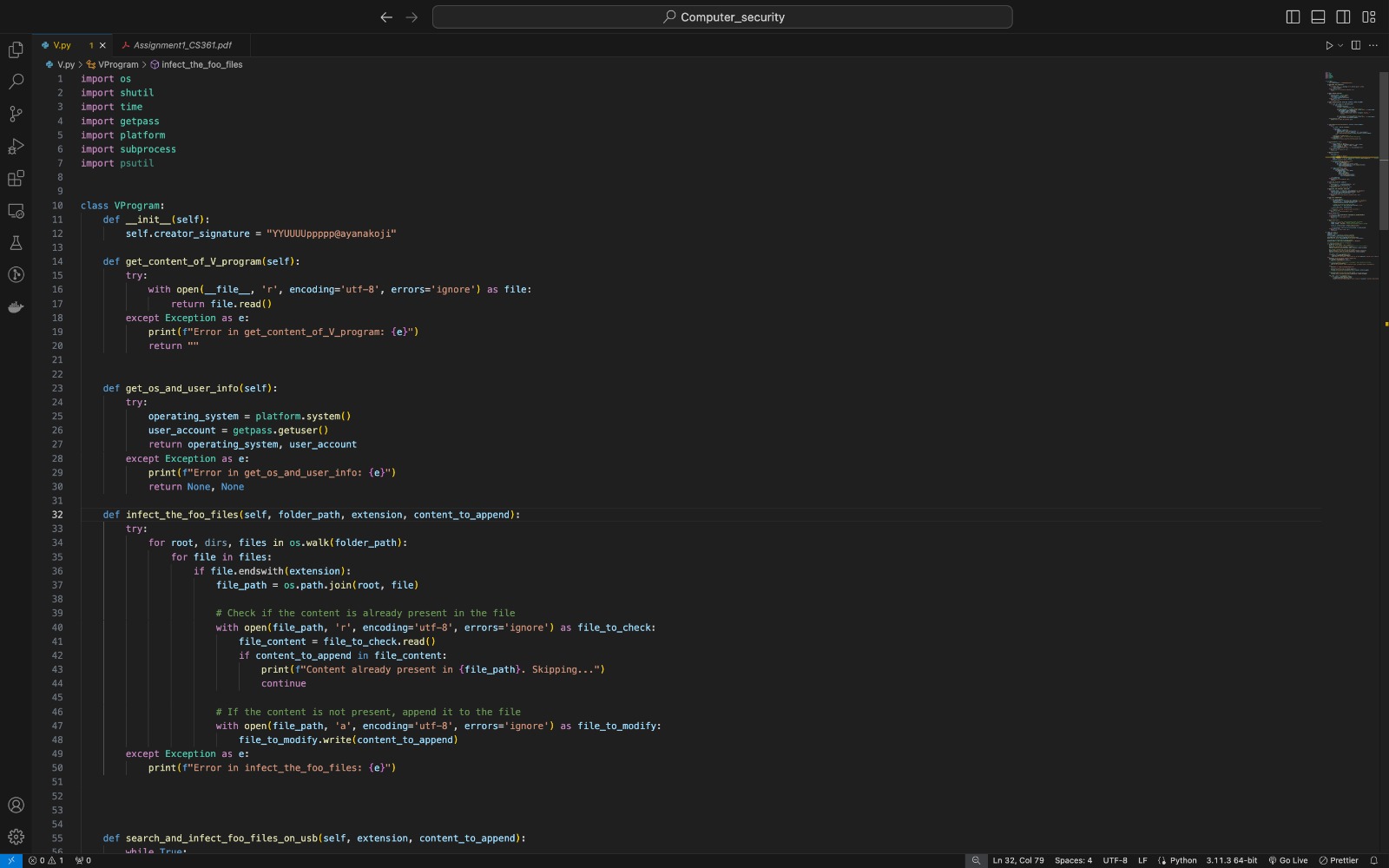
## **Propagation Mechanism:** Spread 'V' by infecting files on USB drives and creating a copy of itself on the removable media.

## **Cross-Computer Propagation:** Spread 'V' to new computers when the infected USB drive is inserted. Specifically, analyse how it identifies a specified directory in the documents folder and copies itself to that location.

## **Antivirus Detection and Response:** We have to observe the effectiveness of antivirus programs in detecting and mitigating the impact of 'V'. Evaluate the ability of antivirus software to identify and quarantine the threat posed by program 'V' during its execution and propagation.

# **Code Analysis**

**a. Searches for files with .foo extension in the documents folder of the user account. Modifies these files by appending the content of 'V' to infect them.**

****

* **Implementation in Code:**
  + The infect\_the\_foo\_files method iterates through the files in the specified folder (documents folder) and checks if they have a '.foo' extension.
  + If a file matches the criteria, 'V's content is appended to that file, effectively infecting it.
* **Explanation:**
  + The code uses the os.walk function to traverse through the directory structure and locate files with the specified extension ('.foo').
  + For each eligible file found, it checks if the content of 'V' is already present to avoid redundant modifications.
  + If the content is not present, it opens the file in append mode and adds the content of 'V', thus infecting the file with the 'V' program.
* **Actions taken by Antivirus:**

Since this operation is occurring within our system, Antivirus software is not taking any action on appending contents of the program to ‘.foo’ files. So it ignores the changes done to ‘.foo’ file.

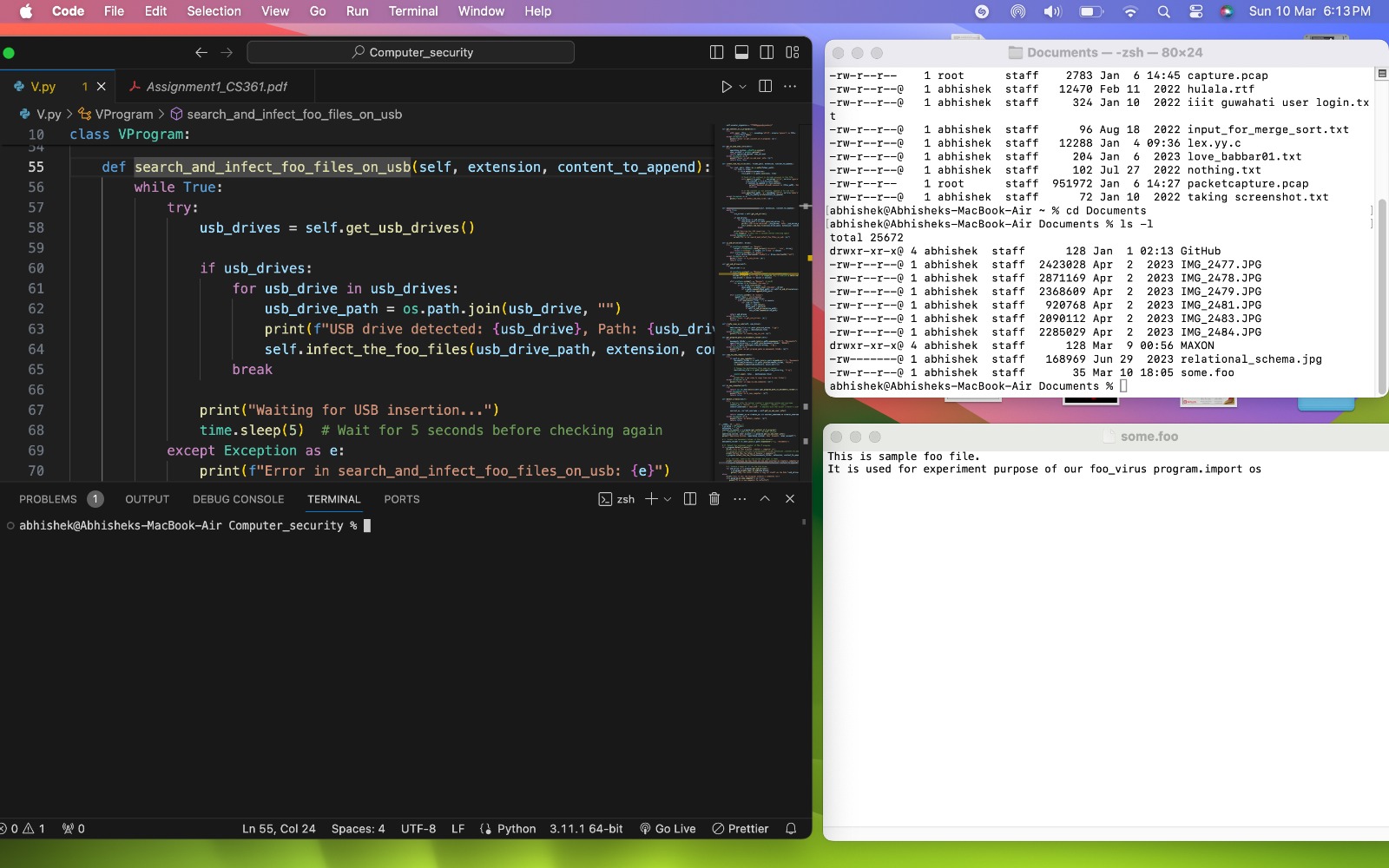


Fig. 1.1 some.foo file before executing program ‘V’

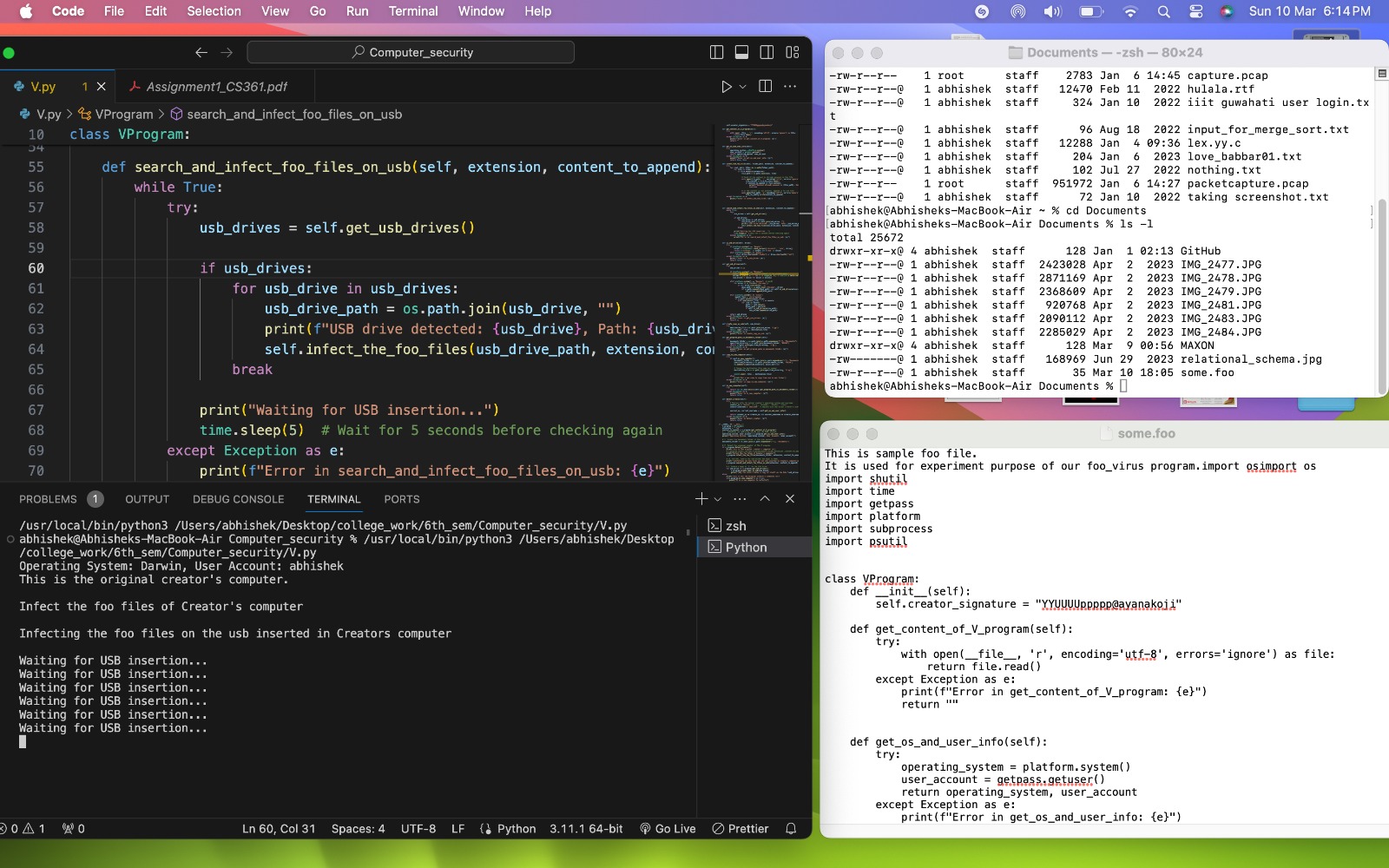
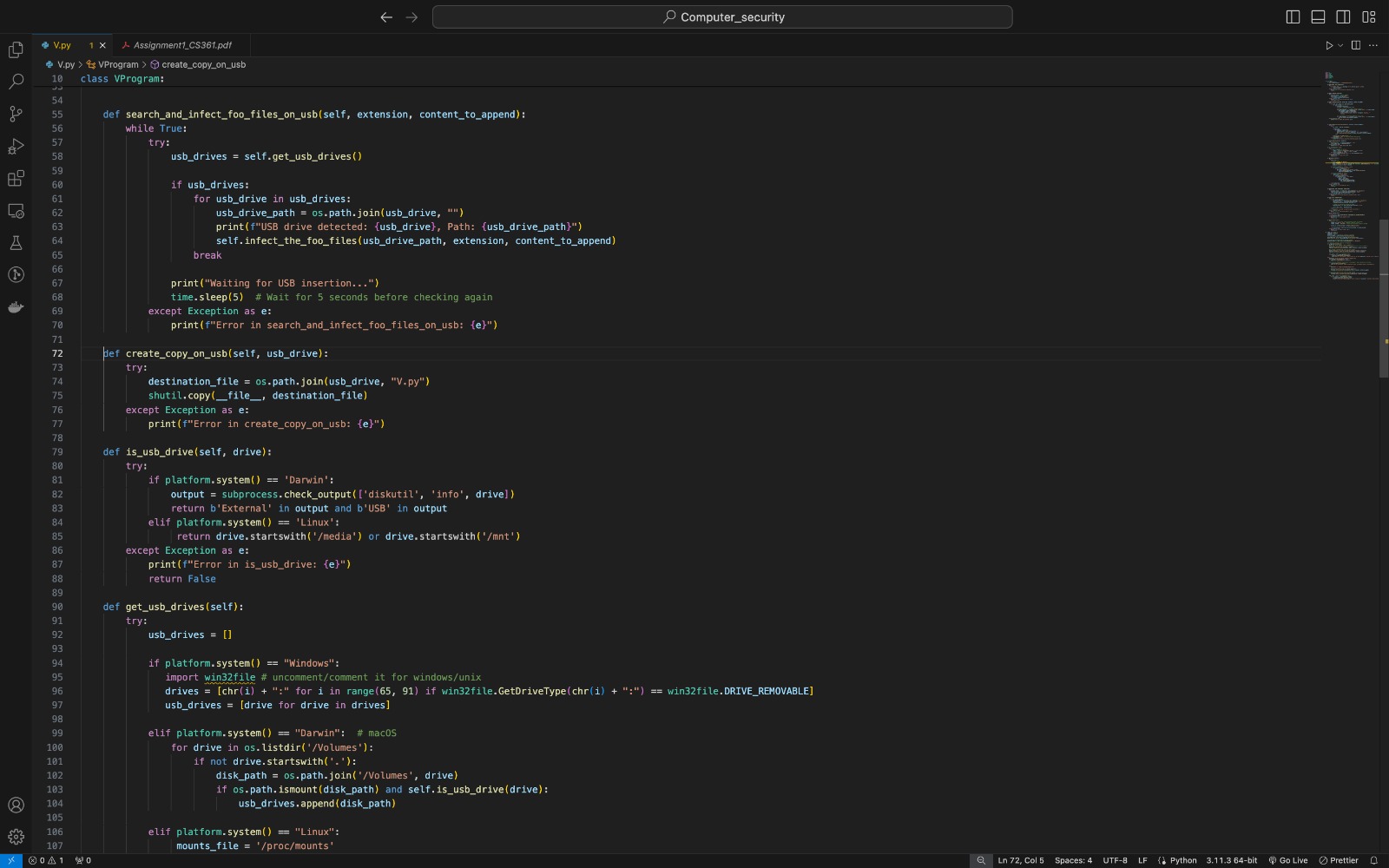


Fig. 1.2 some.foo file after executing program ‘V’

**b. Further, it searches for any mounted USB drives for files with .foo extension and also infects them. It also creates a copy of 'V' on the USB.**

****

* **Implementation in Code:**
  + The search\_and\_infect\_foo\_files\_on\_usb method continuously checks for connected USB drives using the get\_usb\_drives function.And it checks if the usb\_drive is removable or mountable using is\_usb\_drive method.
  + For each detected USB drive, it infects files with the '.foo' extension on the usb drive using the infect\_the\_foo\_files method.
  + Additionally, it creates a copy of the 'V' program on the USB drive using the create\_copy\_on\_usb method.
* **Explanation:**
  + The code utilises the get\_usb\_drives function to obtain a list of connected USB drives based on each platform(Mac/Windows/Linux).
  + For each USB drive found, it infects files with the '.foo' extension using the same mechanism as described in point 'a'.
  + It also creates a copy of 'V' on the USB drive by calling the create\_copy\_on\_usb method, extending the potential spread of the 'V' program.
* **Actions taken by Antivirus:**

For similar reasons as in part ‘a’, no action is taken by the antivirus when the code is copied to the USB drive.

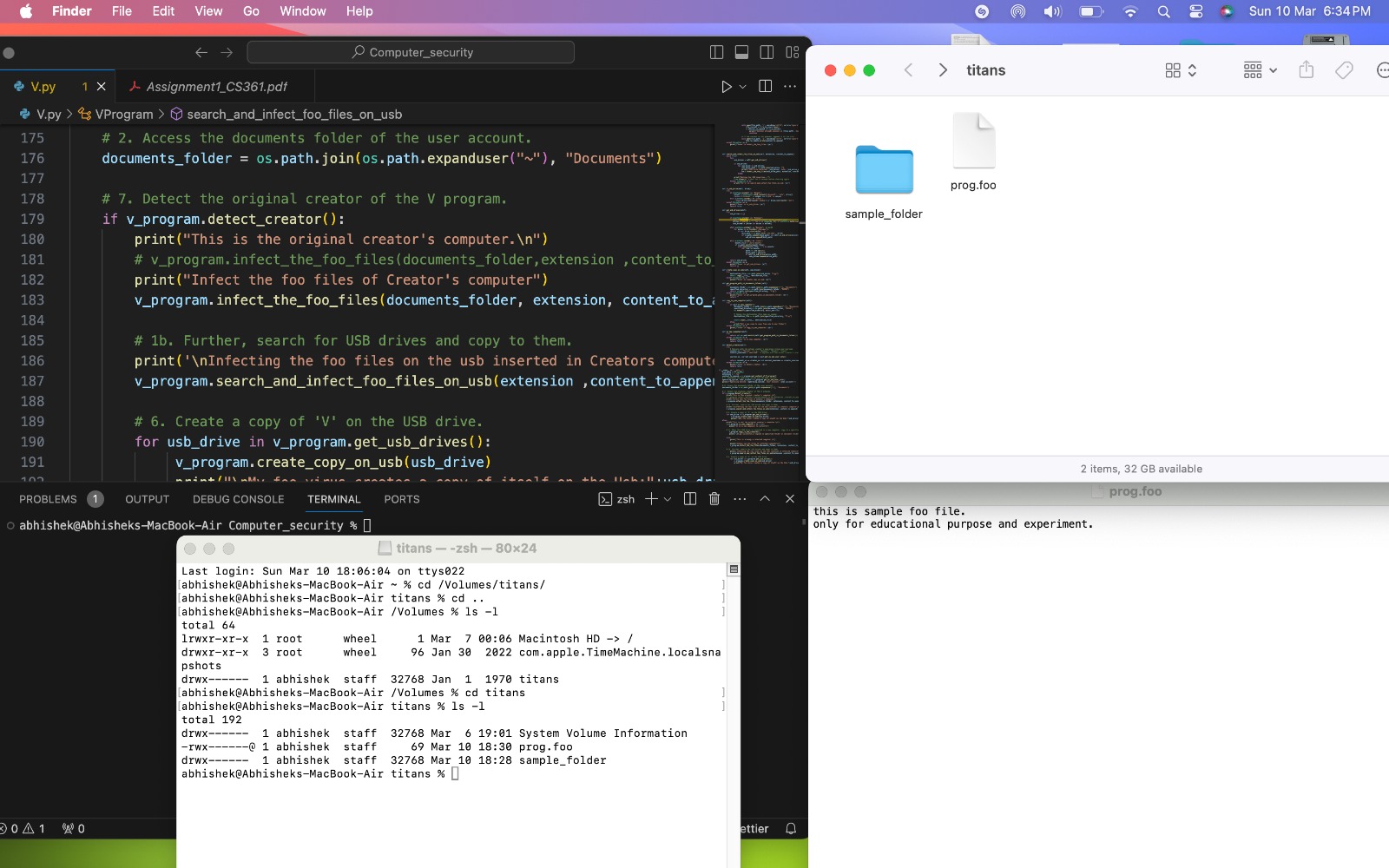


Fig. 2.1 USB Drive and its .foo files before executing program ‘V’

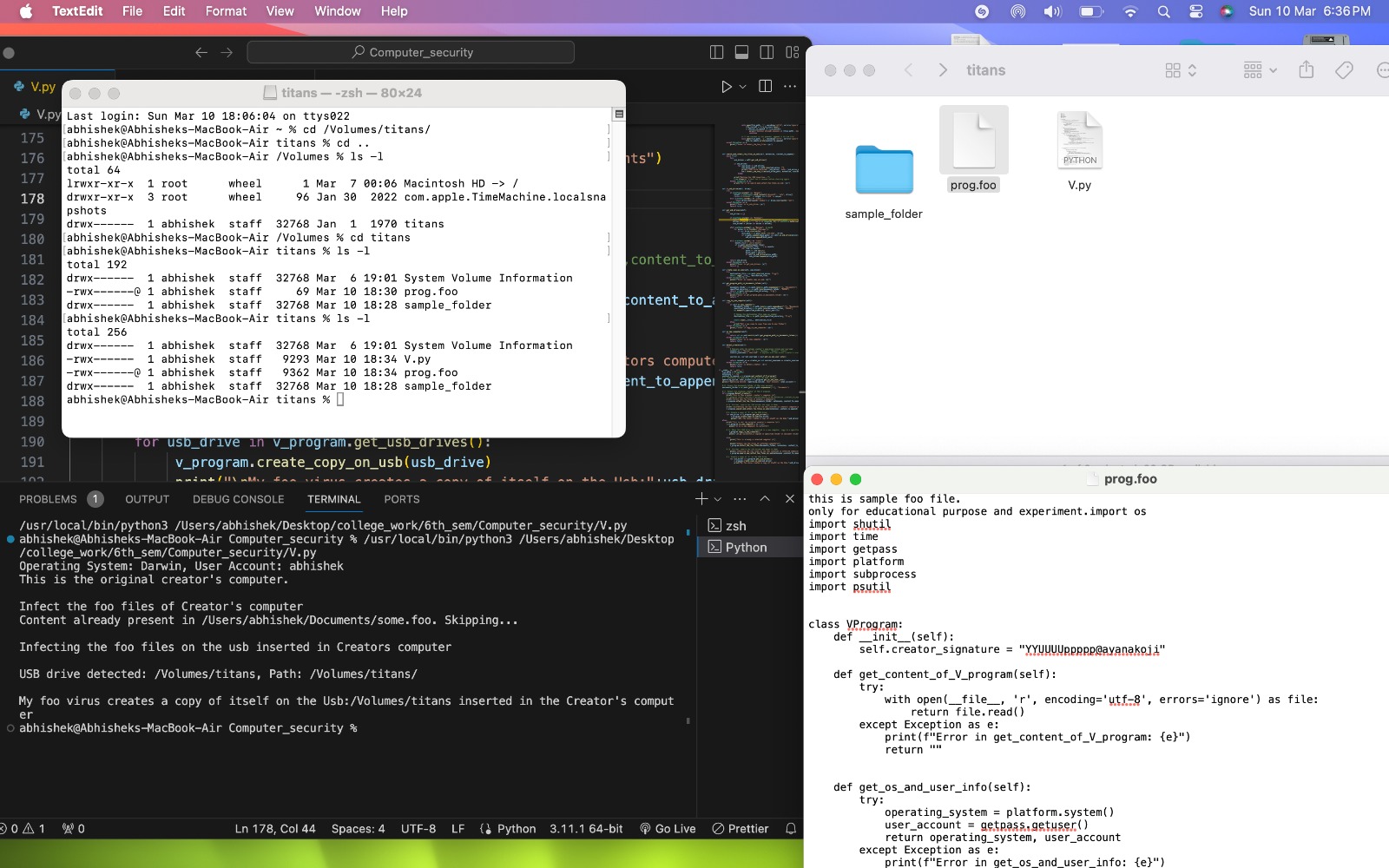


Fig. 2.2 USB Drive and its .foo files after executing program ‘V’

**c. When this USB drive is inserted in a new computer, 'V' searches for a specified directory in the documents folder and copies itself to that directory of the computer.**

****

* **Implementation in Code:**
  + The copy\_to\_new\_computer method checks if the current computer is considered new (based on the absence of 'V' in a specified directory within the documents folder).
  + If it is a new computer, 'V' is copied to the specified directory, potentially infecting the new system.
* **Explanation:**
  + The code checks if the current computer is new by verifying the absence of 'V' in a specified directory within the documents folder.
  + If the computer is determined to be new, it creates the specified directory if it doesn't exist and copies 'V' to that location, potentially spreading the 'V' program to the new computer.
* **Actions taken by Antivirus:**

The antivirus tries to interrupt, block the actions of the ‘.exe’ file and shows an alert message. It first quarantines and then asks for permissions. If allowed, it then executes.

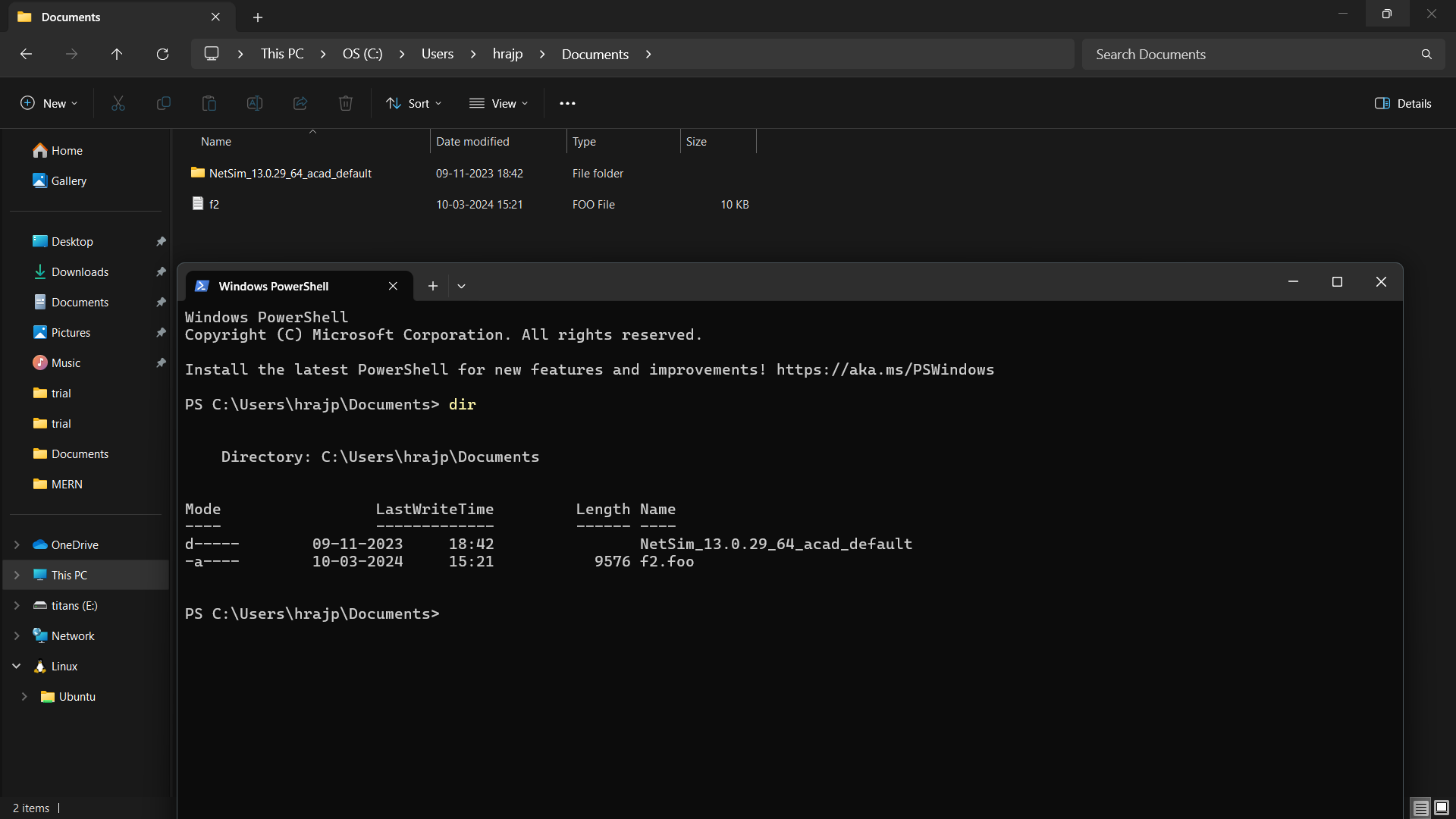
****

Fig. 3.1 New computer’s documents folder before executing program ‘V’

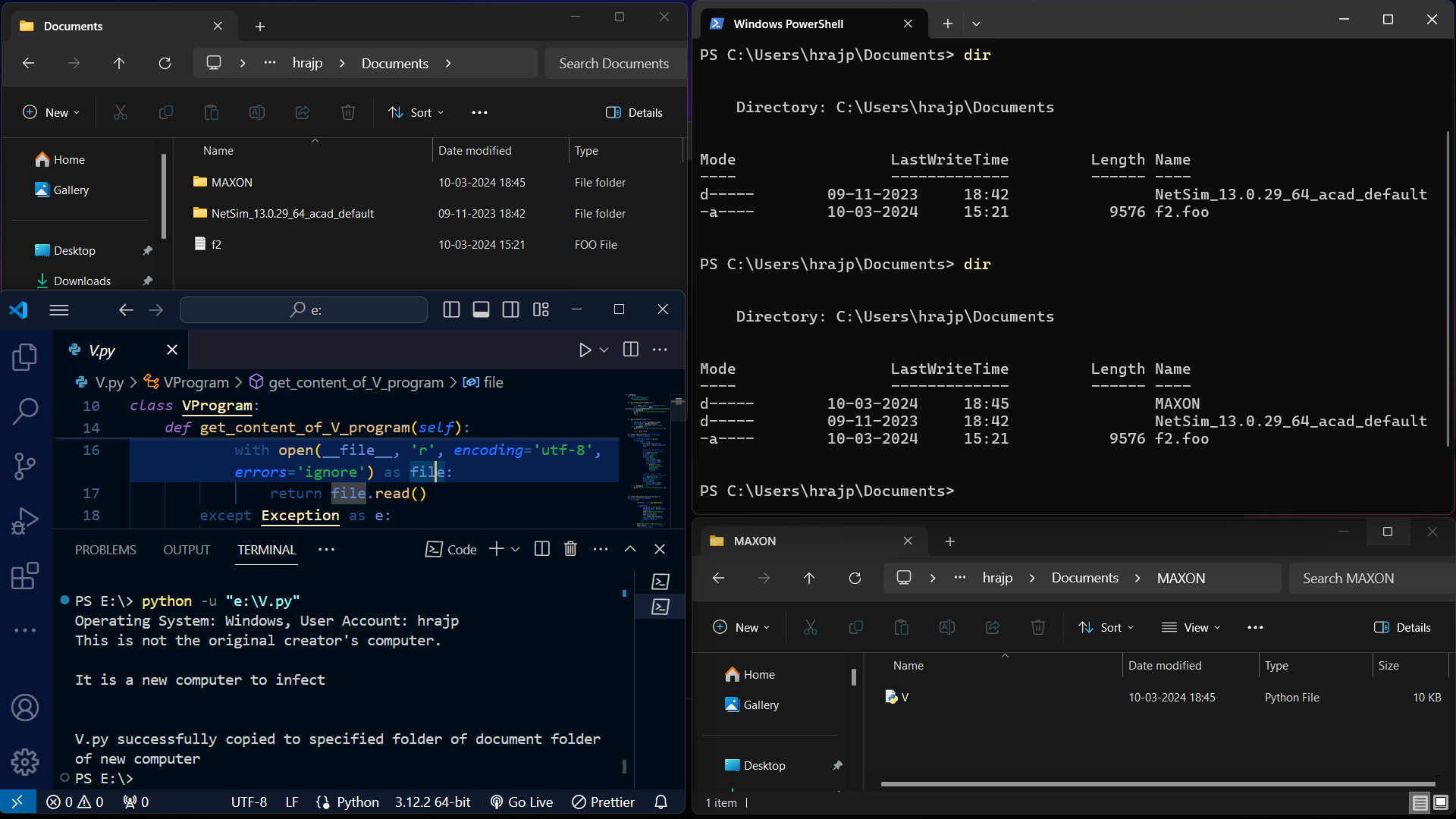
****

Fig. 3.2 New computer’s documents folder after executing program ‘V’

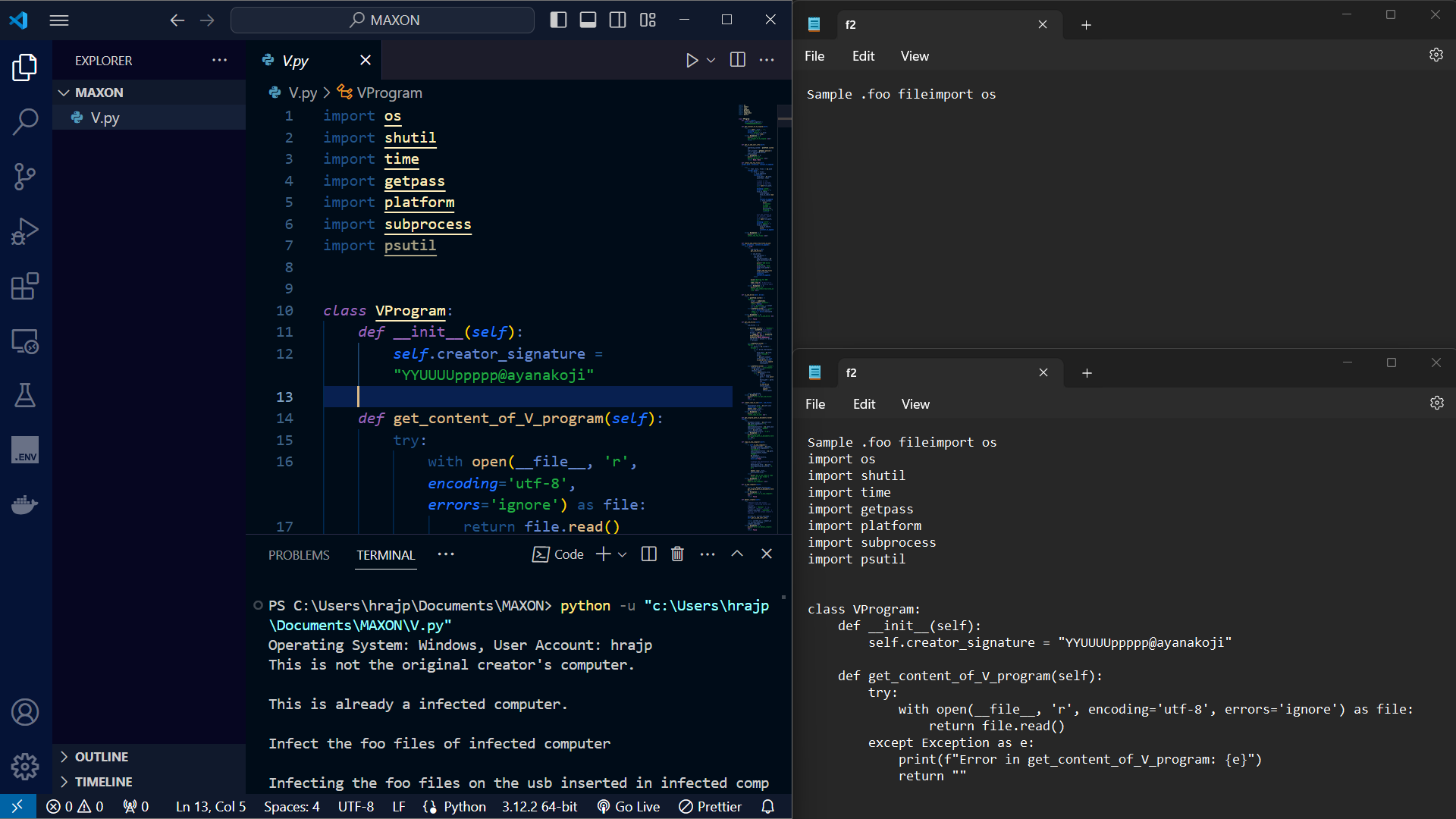


Fig. 3.3 New computer’s .foo files before and after executing program ‘V’

# **Conversion into executable file**

Converting the 'V.py' script into an executable (.exe) file using PyInstaller can increase its vulnerability to execution by any type of user, including those without technical knowledge. This is because:

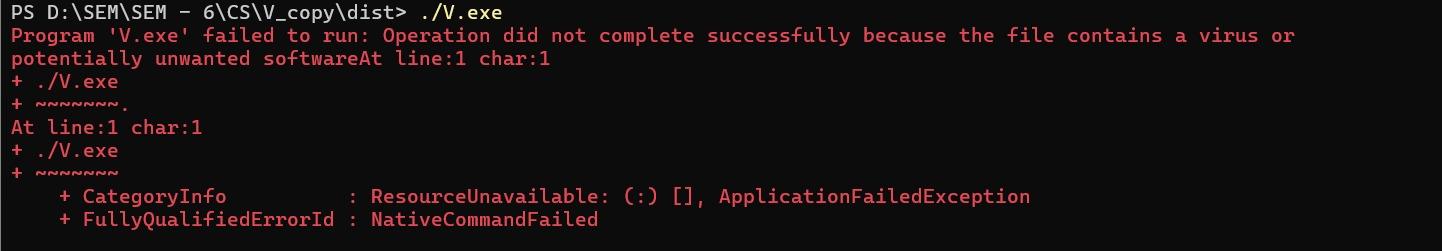
**Reduced Visibility:** Executables obfuscate the underlying Python code, making it harder for users to understand the program's behaviour and potential risks.

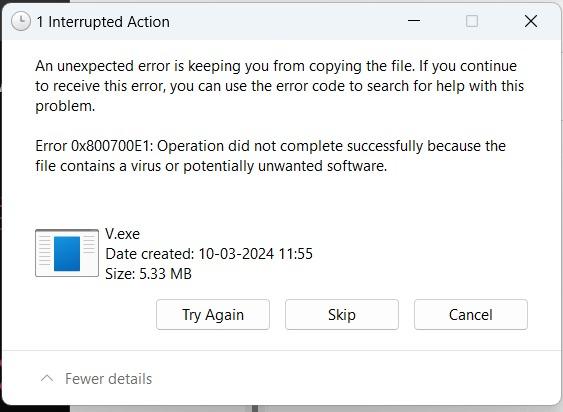
**Simplified Execution:** Executables are easier to run than scripts, increasing the likelihood of users inadvertently executing malicious programs.

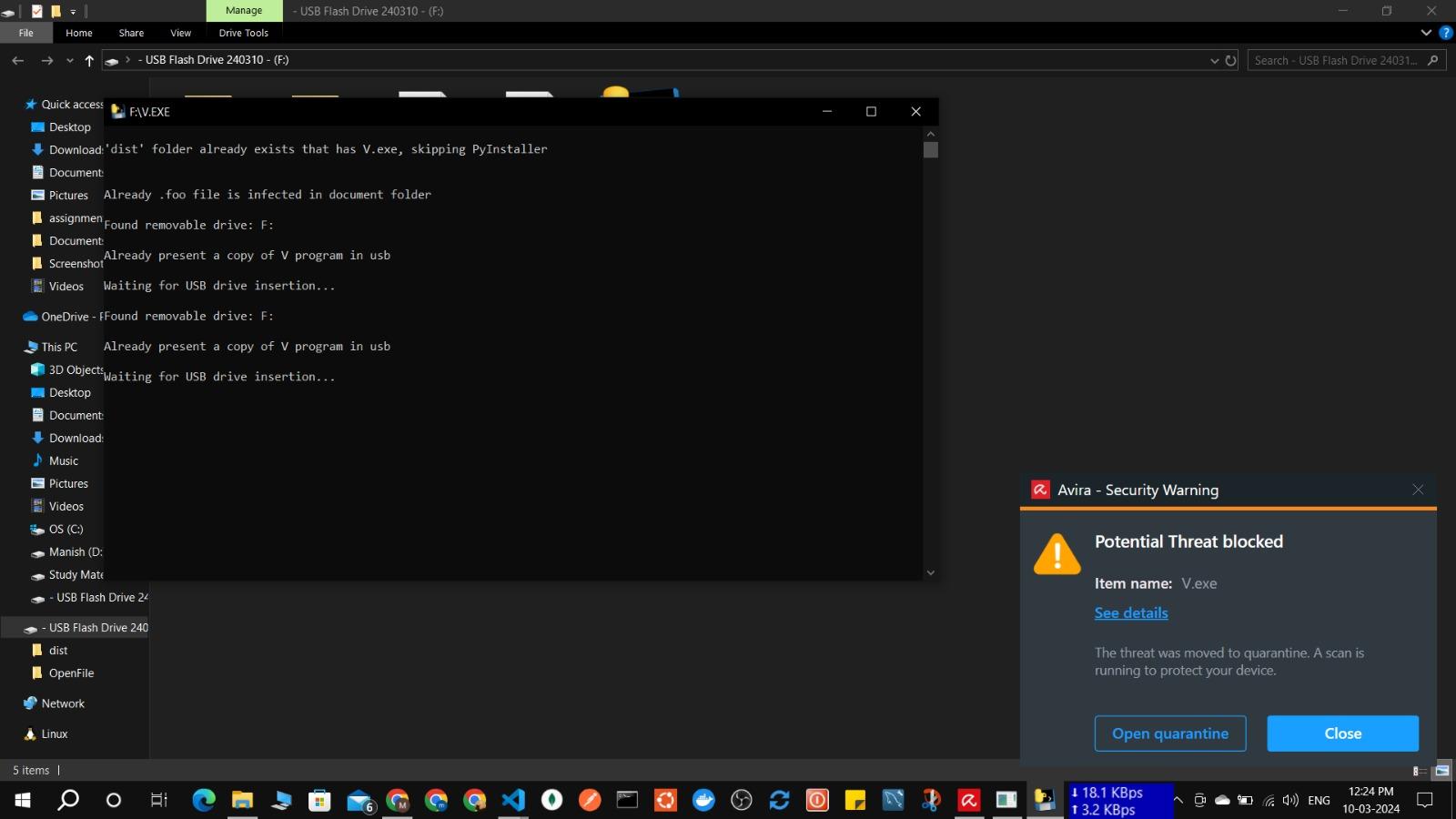
**Increased Distribution Potential:** Executables can be easily distributed, making it easier for attackers to spread malware or viruses disguised as legitimate programs.

**d. Comment on the actions taken by the antivirus program**

**present in the computer.**







**Interpretation:** This error message indicates that the antivirus software has detected a potential threat in the "V.exe" file and is preventing it from being copied.

The error code 0x800700E1 specifically suggests that the file contains a virus or potentially unwanted software (PUA).

**Antivirus Actions:**

Quarantine or Deletion: The antivirus software likely quarantined or deleted the "V.exe" file to prevent it from harming the system.

Real-Time Protection: The antivirus is actively monitoring the system and taking action to block potential threats in real time.

# **Conclusion**

* + This assignment has provided valuable insights into the behaviour and potential impact of the foo\_virus (V.py/V.exe) malware. We have explored its capabilities, including its ability to search for and modify files with the .foo extension, as well as its potential to spread through USB drives.
  + ‘V.exe’ is a potentially harmful malware: Its ability to modify files and spread through USB drives can lead to data loss, system instability, and other security risks.
  + In conclusion, the analysis of the V.exe script emphasises the importance of cybersecurity awareness and proactive measures to mitigate risks associated with malware propagation and exploitation. Understanding such threats is crucial for safeguarding digital assets and privacy in both personal and organisational contexts.

# **Link to our Code:** [google\_drive\_link](https://drive.google.com/file/d/1nz7igMZBC3n9u28Qqp6srE69N6t1Gwk2/view?usp=sharing)