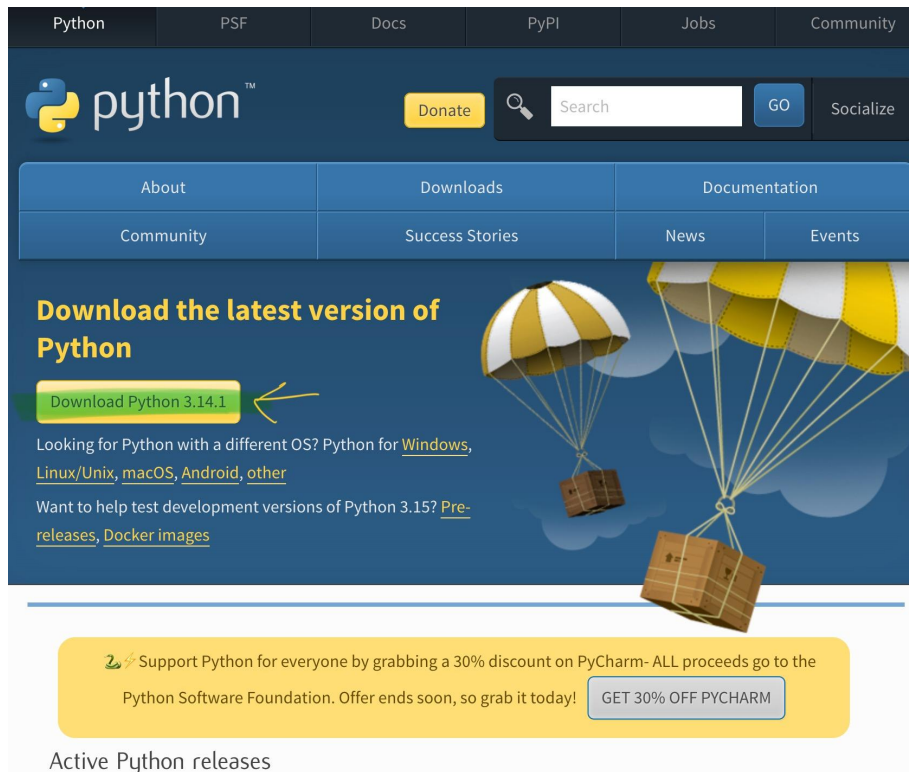


USB Drop Attack Simulation Installation Guide

This installation guide describes the installation of the USB Drop Attack simulation, explaining how to install the required tools and build the executable (.exe) from the Python script.

1. Install Python

Download and install Python 3.14.0 or latest version from the official Python website. During installation, make sure that the checkbox “Add Python to PATH” is selected. Complete the setup using the default installation settings.



During installation, make sure that the checkbox “Add Python to PATH” is selected. Complete the setup using the default installation settings.

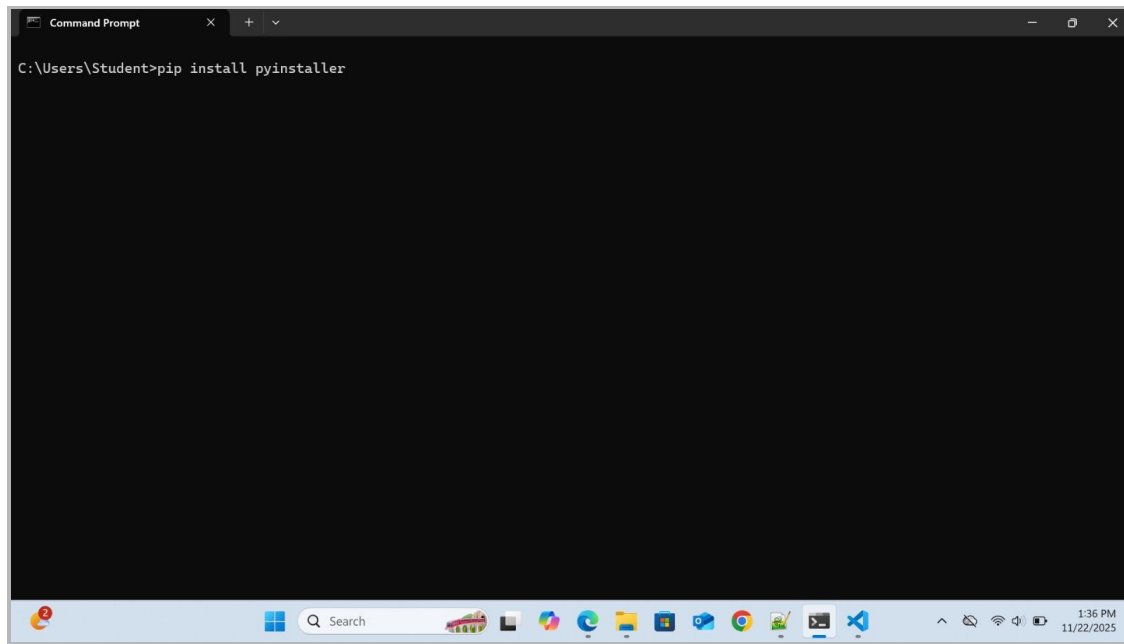
2. Install Required Python Module PyInstaller

PyInstaller is used to convert the Python script into a Windows executable.

Install it using:

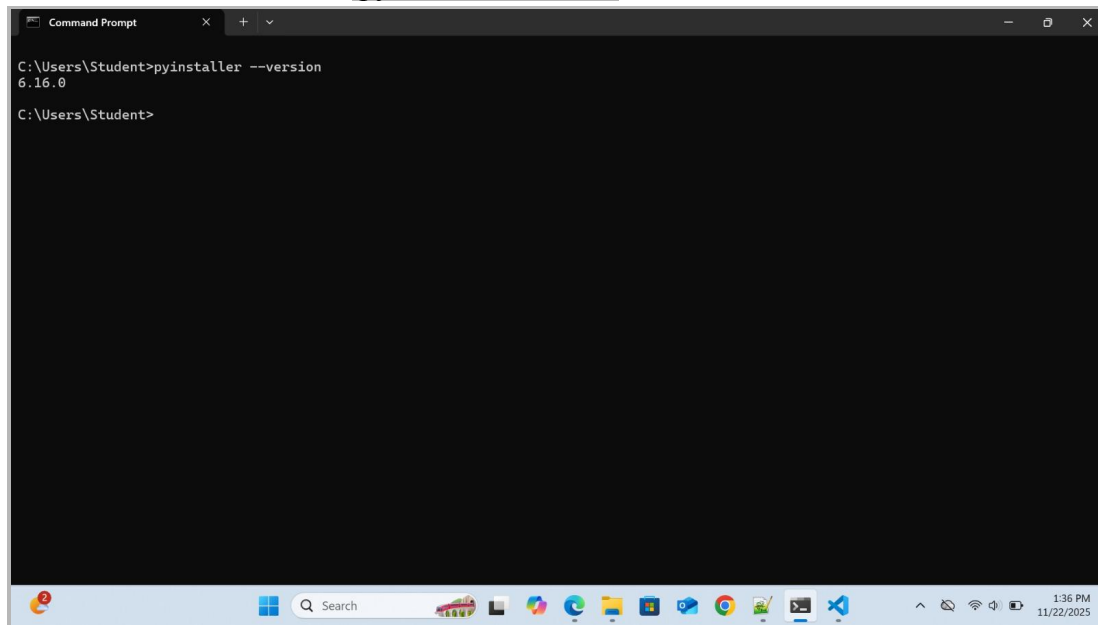
```
pip install pyinstaller
```

Open Command Prompt and install the required external module: `pip install pyinstaller`



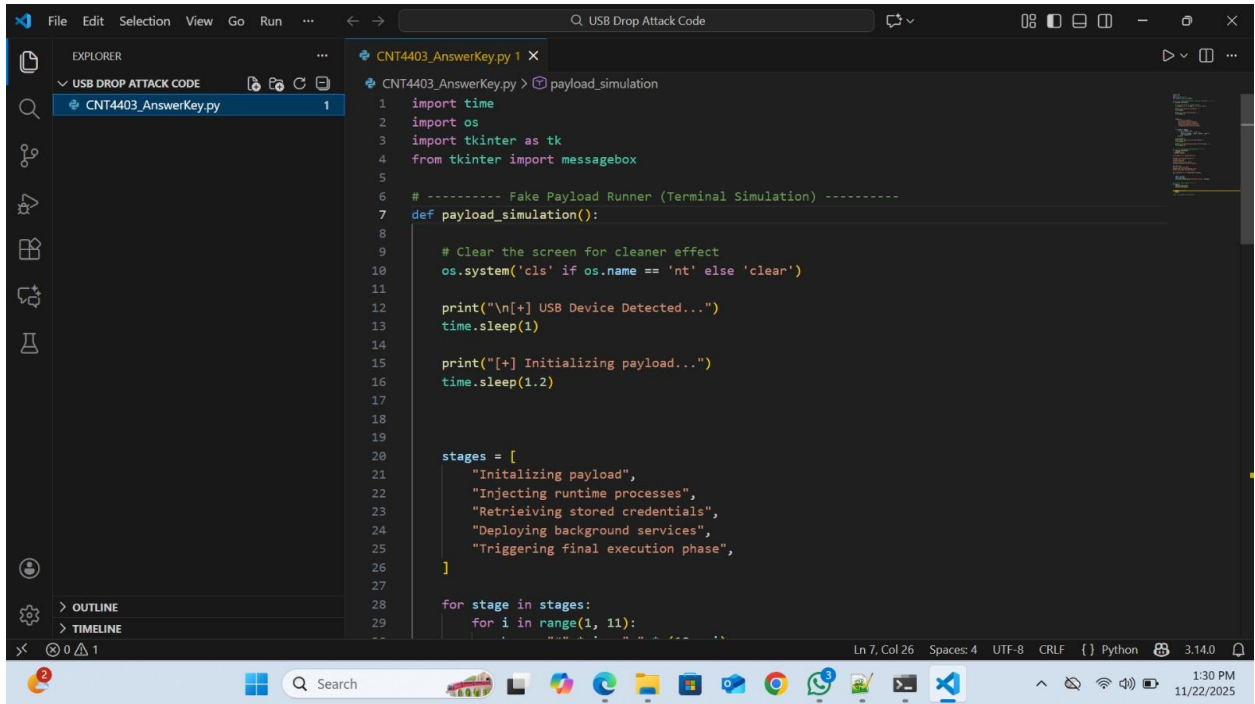
```
Command Prompt
C:\Users\Student>pip install pyinstaller
```

Confirm it is installed with `pyinstaller --version`



```
Command Prompt
C:\Users\Student>pyinstaller --version
6.16.0
C:\Users\Student>
```

Step 3: Before converting the script, run it in a Python environment to confirm it executes correctly without errors in Visual Studio Code.



```
1 import time
2 import os
3 import tkinter as tk
4 from tkinter import messagebox
5
6 # ----- Fake Payload Runner (Terminal Simulation) -----
7 def payload_simulation():
8
9     # Clear the screen for cleaner effect
10    os.system('cls' if os.name == 'nt' else 'clear')
11
12    print("\n[+] USB Device Detected...")
13    time.sleep(1)
14
15    print("[+] Initializing payload...")
16    time.sleep(1.2)
17
18
19
20    stages = [
21        "Initializing payload",
22        "Injecting runtime processes",
23        "Retrieving stored credentials",
24        "Deploying background services",
25        "Triggering final execution phase",
26    ]
27
28    for stage in stages:
29        for i in range(1, 11):
30            print(f"[+] {stage} (Progress: {i}/10) ...")
31            time.sleep(0.5)
```

You should see the fake terminal output beginning and the Tkinter popup appearing.

insert screenshot from moath

Step 4: Convert PythonScript to Executable (.exe)

Navigate to the directory where your Python script is stored and run the following command:

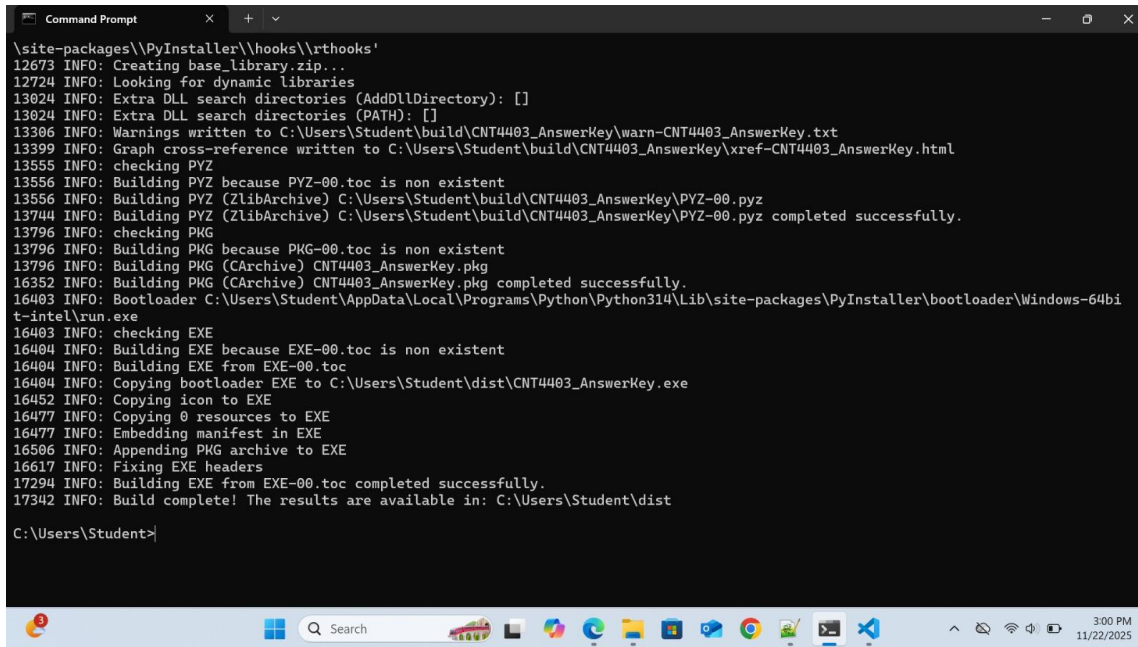
```
pyinstaller --onefile --windowed CNT4403_AnswerKey.py
```

--onfile bundles everything into a single executable

--windowed prevents a secondary console window if not wanted

Pyinstaller will then generate two folders: build/ and dist/.

The final executable will be located inside the dist/ folder.



```
\site-packages\PyInstaller\hooks\rthooks'
12673 INFO: Creating base_library.zip...
12724 INFO: Looking for dynamic libraries
13024 INFO: Extra DLL search directories (AddDllDirectory): []
13024 INFO: Extra DLL search directories (PATH): []
13306 INFO: Warnings written to C:\Users\Student\build\CNT4403_AnswerKey\warn-CNT4403_AnswerKey.txt
13399 INFO: Graph cross-reference written to C:\Users\Student\build\CNT4403_AnswerKey\xref-CNT4403_AnswerKey.html
13555 INFO: checking PYZ
13556 INFO: Building PYZ because PYZ-00.toc is non existent
13556 INFO: Building PYZ (ZLibArchive) C:\Users\Student\build\CNT4403_AnswerKey\PYZ-00.pyz
13744 INFO: Building PYZ (ZLibArchive) C:\Users\Student\build\CNT4403_AnswerKey\PYZ-00.pyz completed successfully.
13796 INFO: checking PKG
13796 INFO: Building PKG because PKG-00.toc is non existent
13796 INFO: Building PKG (CArchive) CNT4403_AnswerKey.pkg
16352 INFO: Building PKG (CArchive) CNT4403_AnswerKey.pkg completed successfully.
16403 INFO: Bootloader C:\Users\Student\AppData\Local\Programs\Python\Python314\Lib\site-packages\PyInstaller\bootloader\Windows-64bit-intel\run.exe
16403 INFO: checking EXE
16404 INFO: Building EXE because EXE-00.toc is non existent
16404 INFO: Building EXE from EXE-00.toc
16404 INFO: Copying bootloader EXE to C:\Users\Student\dist\CNT4403_AnswerKey.exe
16452 INFO: Copying icon to EXE
16477 INFO: Copying 0 resources to EXE
16477 INFO: Embedding manifest in EXE
16506 INFO: Appending PKG archive to EXE
16617 INFO: Fixing EXE headers
17294 INFO: Building EXE from EXE-00.toc completed successfully.
17342 INFO: Build complete! The results are available in: C:\Users\Student\dist

C:\Users\Student>
```

Terminal will show “Build complete!” when the executable is ready.

Step 5: Test the Executable

After the PyInstaller finishes, open the dist/ folder and double -click the generated .exe to test the application.

Verify

- Fake terminal simulation works
- GUI warning pop-up displays
- Program closes cleanly

insert screenshot

Step 6: Distribute the Executable

The .exe file can now be dragged into a USB drive and shared with users.

For any questions please contact the Product Manager.