import tkinter as tk

from tkinter import scrolledtext

import subprocess

#NMAP

import subprocess

def run\_nmap\_scan(target\_ip):

try:

# Running the Nmap command

result = subprocess.run(["nmap", "-F", target\_ip], capture\_output=True, text=True, check=True)

# Returning the output

return result.stdout

except subprocess.CalledProcessError as e:

return f"An error occurred: {e}\nOutput: {e.output}"

scan\_result = run\_nmap\_scan("192.168.1.1") # replace with your target IP

print(scan\_result)

#Steghide tool:

import subprocess

def embed\_data(cover\_file, data\_file, passphrase):

try:

result = subprocess.run(["steghide", "embed", "-cf", cover\_file, "-ef", data\_file, "-p", passphrase], capture\_output=True, text=True, check=True)

return "Data embedded successfully.\n" + result.stdout

except subprocess.CalledProcessError as e:

return f"An error occurred: {e}\nOutput: {e.output}"

def extract\_data(cover\_file, passphrase):

try:

result = subprocess.run(["steghide", "extract", "-sf", cover\_file, "-p", passphrase], capture\_output=True, text=True, check=True)

return "Data extracted successfully.\n" + result.stdout

except subprocess.CalledProcessError as e:

return f"An error occurred: {e}\nOutput: {e.output}"

# Example use

embed\_result = embed\_data("/home/bharanisoundhar/Downloads/pict.jpeg", "/home/bharanisoundhar/text", "your\_password")

print(embed\_result)

# extract\_result = extract\_data("/home/bharanisoundhar/Downloads/pict.jpeg", "your\_password")

# print(extract\_result)

#Encryption/decryption tool:

import subprocess

def encrypt\_file(file\_path, passphrase):

try:

result = subprocess.run(["gpg", "--batch", "--yes", "--passphrase", passphrase, "-c", file\_path], capture\_output=True, text=True, check=True)

return "File encrypted successfully.\n" + result.stdout

except subprocess.CalledProcessError as e:

return f"An error occurred: {e}\nOutput: {e.output}"

def decrypt\_file(file\_path, passphrase):

try:

encrypted\_file = file\_path + ".gpg"

result = subprocess.run(["gpg", "--batch", "--yes", "--passphrase", passphrase, "-o", file\_path, "-d", encrypted\_file], capture\_output=True, text=True, check=True)

return "File decrypted successfully.\n" + result.stdout

except subprocess.CalledProcessError as e:

return f"An error occurred: {e}\nOutput: {e.output}"

# Example use

# Replace \_passphrase' with your actual passphrase

encrypt\_result = encrypt\_file("/home/bharanisoundhar/encryption", "your\_passphrase")

print(encrypt\_result)

# To decrypt, uncomment the lines below and comment out the encryption lines

# decrypt\_result = decrypt\_file("/home/bharanisoundhar/encryption", "your\_passphrase")

# print(decrypt\_result)

#Simple firewall:

import subprocess

def enable\_firewall():

try:

subprocess.run(["sudo", "ufw", "enable"], check=True)

return "Firewall enabled."

except subprocess.CalledProcessError as e:

return f"Error enabling firewall: {e}"

def disable\_firewall():

try:

subprocess.run(["sudo", "ufw", "disable"], check=True)

return "Firewall disabled."

except subprocess.CalledProcessError as e:

return f"Error disabling firewall: {e}"

def add\_rule(rule):

try:

subprocess.run(["sudo", "ufw"] + rule.split(), check=True)

return f"Rule added: {rule}"

except subprocess.CalledProcessError as e:

return f"Error adding rule '{rule}': {e}"

def remove\_rule(rule):

try:

subprocess.run(["sudo", "ufw", "delete"] + rule.split(), check=True)

return f"Rule removed: {rule}"

except subprocess.CalledProcessError as e:

return f"Error removing rule '{rule}': {e}"

# Example usage

print(enable\_firewall())

print(add\_rule("allow 22"))

print(add\_rule("deny from 192.168.1.5"))

print(remove\_rule("allow 22"))

print(disable\_firewall())

#simple IDS:

import subprocess

def start\_suricata(interface="eth0"):

try:

subprocess.run(["sudo", "suricata", "-c", "/etc/suricata/suricata.yaml", "-i", interface], check=True)

return "Suricata started on interface " + interface

except subprocess.CalledProcessError as e:

return f"Error starting Suricata: {e}"

def stop\_suricata():

try:

subprocess.run(["sudo", "pkill", "suricata"], check=True)

return "Suricata stopped."

except subprocess.CalledProcessError as e:

return f"Error stopping Suricata: {e}"

# Example usage

print(start\_suricata("eth0")) # Replace "eth0" with your network interface

# print(stop\_suricata())

# Initialize the main window

root = tk.Tk()

root.title("CyberGuard Toolkit")

# Function to update output text area

def update\_output(message):

output\_text.insert(tk.END, message + "\n")

output\_text.see(tk.END)

# Nmap Section

tk.Label(root, text="Nmap - Enter Target IP:").pack()

nmap\_ip\_entry = tk.Entry(root)

nmap\_ip\_entry.pack()

nmap\_button = tk.Button(root, text="Run Nmap Scan", command=lambda: update\_output(run\_nmap\_scan(nmap\_ip\_entry.get())))

nmap\_button.pack()

# Steghide Section

tk.Label(root, text="Steghide - Cover File, Data File, Passphrase:").pack()

steghide\_cover\_entry = tk.Entry(root)

steghide\_cover\_entry.pack()

steghide\_data\_entry = tk.Entry(root)

steghide\_data\_entry.pack()

steghide\_pass\_entry = tk.Entry(root, show="\*")

steghide\_pass\_entry.pack()

steghide\_button = tk.Button(root, text="Embed Data", command=lambda: update\_output(embed\_data(steghide\_cover\_entry.get(), steghide\_data\_entry.get(), steghide\_pass\_entry.get())))

steghide\_button.pack()

# Encryption/Decryption Section

tk.Label(root, text="Encryption - File Path, Passphrase:").pack()

encryption\_file\_entry = tk.Entry(root)

encryption\_file\_entry.pack()

encryption\_pass\_entry = tk.Entry(root, show="\*")

encryption\_pass\_entry.pack()

encryption\_button = tk.Button(root, text="Encrypt File", command=lambda: update\_output(encrypt\_file(encryption\_file\_entry.get(), encryption\_pass\_entry.get())))

encryption\_button.pack()

# Firewall Section

firewall\_button = tk.Button(root, text="Enable Firewall", command=lambda: update\_output(enable\_firewall()))

firewall\_button.pack()

firewall\_off\_button = tk.Button(root, text="Disable Firewall", command=lambda: update\_output(disable\_firewall()))

firewall\_off\_button.pack()

# Suricata Section

tk.Label(root, text="Suricata - Enter Interface (e.g., eth0):").pack()

suricata\_interface\_entry = tk.Entry(root)

suricata\_interface\_entry.pack()

suricata\_button = tk.Button(root, text="Start Suricata", command=lambda: update\_output(start\_suricata(suricata\_interface\_entry.get())))

suricata\_button.pack()

suricata\_stop\_button = tk.Button(root, text="Stop Suricata", command=lambda: update\_output(stop\_suricata()))

suricata\_stop\_button.pack()

# Output Text Area

output\_text = scrolledtext.ScrolledText(root, height=10, width=50)

output\_text.pack()

# Start the GUI event loop

root.mainloop()