import re

import random

import string

def perform\_vulnerability\_scan(target):

open\_ports = check\_open\_ports(target)

software\_versions = check\_software\_versions(target)

weak\_configurations = check\_weak\_configurations(target)

report = generate\_vulnerability\_report(open\_ports, software\_versions, weak\_configurations)

print("Vulnerability Report:")

print(report)

def check\_open\_ports(target):

open\_ports = random.sample(range(1, 65536), 10)

return open\_ports

def check\_software\_versions(target):

software\_versions = {

"Apache": "2.4.29",

"MySQL": "5.7.21",

"PHP": "7.2.6"

}

return software\_versions

def check\_weak\_configurations(target):

weak\_configurations = ["Default username and password", "Insecure SSH configuration"]

return weak\_configurations

def generate\_vulnerability\_report(open\_ports, software\_versions, weak\_configurations):

report = f"Open Ports: {open\_ports}\n\n"

report += "Software Versions:\n"

for software, version in software\_versions.items():

report += f"{software}: {version}\n"

report += "\nWeak Configurations:\n"

for configuration in weak\_configurations:

report += f"- {configuration}\n"

return report

def perform\_log\_analysis(log\_file):

log\_entries = parse\_log\_file(log\_file)

suspicious\_activities = analyze\_log\_entries(log\_entries)

report = generate\_log\_analysis\_report(suspicious\_activities)

print("Log Analysis Report:")

print(report)

def parse\_log\_file(log\_file):

log\_entries = [

"2023-06-25 12:30:45 - IP: 192.168.0.10 - Access denied",

"2023-06-25 13:15:20 - IP: 10.0.0.5 - Error: File not found",

"2023-06-25 14:05:55 - IP: 192.168.0.15 - Authentication failed"

]

return log\_entries

def analyze\_log\_entries(log\_entries):

suspicious\_activities = random.sample(log\_entries, 2)

return suspicious\_activities

def generate\_log\_analysis\_report(suspicious\_activities):

report = "Suspicious Activities:\n"

for activity in suspicious\_activities:

report += f"- {activity}\n"

return report

def main():

print("=== PySecAutomation ===")

while True:

print("\nSelect an option:")

print("1. Perform vulnerability scanning")

print("2. Perform log analysis")

print("3. Exit")

choice = input("Enter your choice (1-3): ")

if choice == "1":

target = input("Enter the target IP address or hostname: ")

perform\_vulnerability\_scan(target)

elif choice == "2":

log\_file = input("Enter the path to the log file: ")

perform\_log\_analysis(log\_file)

elif choice == "3":

print("Exiting BoaPython...")

break

else:

print("Invalid choice. Please try again.")

if \_\_name\_\_ == "\_\_main\_\_":

main()

