Name   Mohsin Nawaz

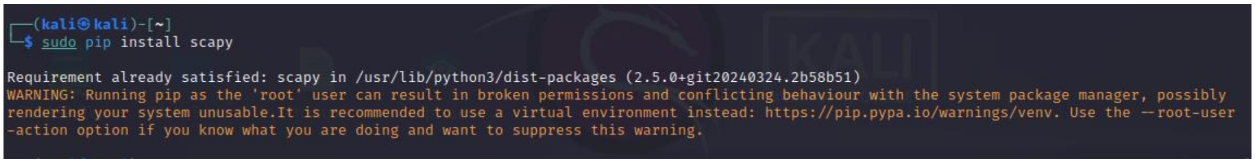
Code Alpha

CYBER SECURITY INTERNSHIP TASKS

BASIC NETWORK SNIFFER : Build a network sniffer in Python that captures and analyzes network traffic. This project will help you understand how data flows on a network and how network packets are structured.

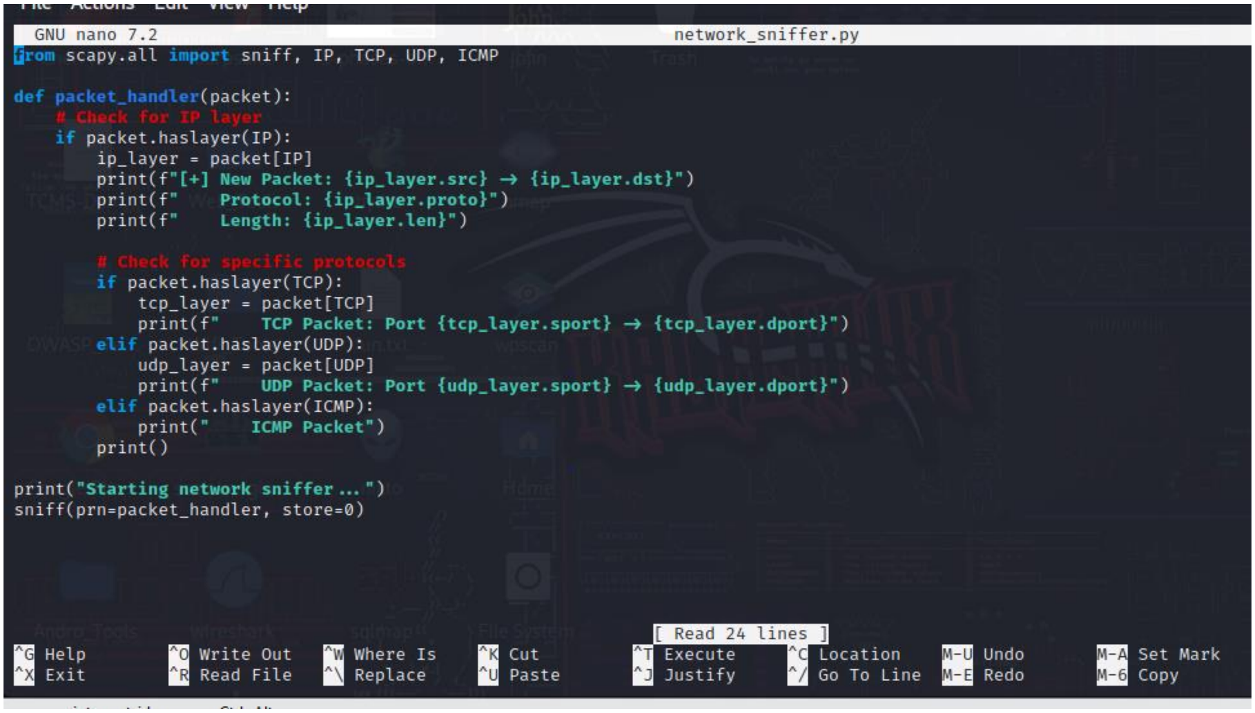
Answer

Installation of scapy libraries:



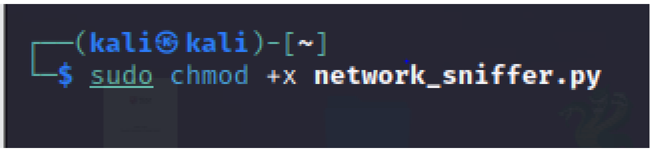
```
┌──(kali㉿kali)-[~]
└─$ sudo pip install scapy

Requirement already satisfied: scapy in /usr/lib/python3/dist-packages (2.5.0+git20240324.2b58b51)
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager, possibly
rendering your system unusable.It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv. Use the --root-user
-action option if you know what you are doing and want to suppress this warning.
```

Creating a python file named network_sniffer.py then open it for coding :



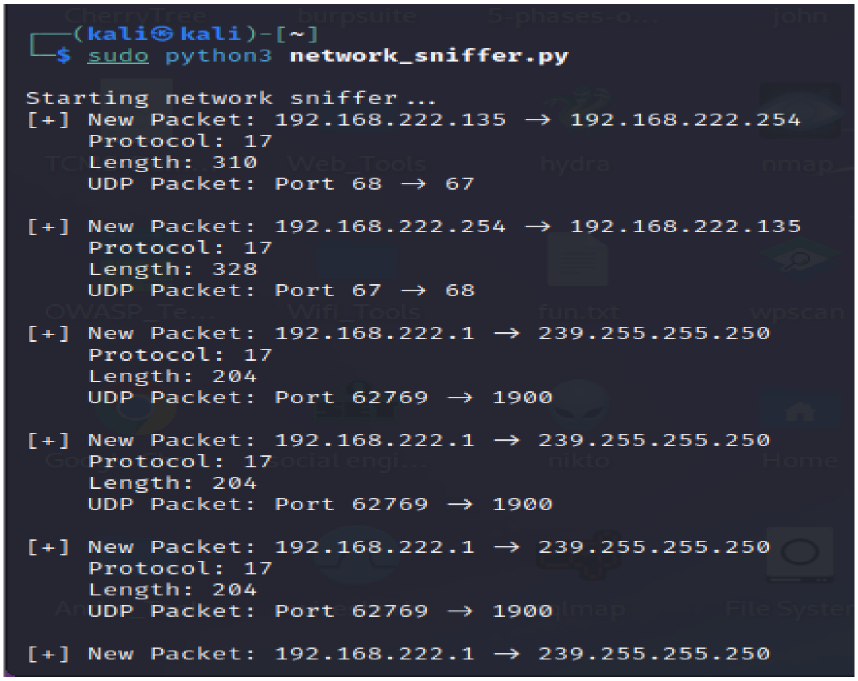
```
  GNU nano 7.2                                          network_sniffer.py
from scapy.all import sniff, IP, TCP, UDP, ICMP

def packet_handler(packet):
    # Check for IP layer
    if packet.haslayer(IP):
        ip_layer = packet[IP]
        print(f"[+] New Packet: {ip_layer.src} → {ip_layer.dst}")
        print(f"    Protocol: {ip_layer.proto}")
        print(f"    Length: {ip_layer.len}")

        # Check for specific protocols
        if packet.haslayer(TCP):
            tcp_layer = packet[TCP]
            print(f"    TCP Packet: Port {tcp_layer.sport} → {tcp_layer.dport}")
        elif packet.haslayer(UDP):
            udp_layer = packet[UDP]
            print(f"    UDP Packet: Port {udp_layer.sport} → {udp_layer.dport}")
        elif packet.haslayer(ICMP):
            print("    ICMP Packet")
        print()

print("Starting network sniffer ...")
sniff(prn=packet_handler, store=0)

                                        [ Read 24 lines ]
^G Help       ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo      M-A Set Mark
^X Exit       ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/ Go To Line M-E Redo      M-6 Copy
```

After that save the file and close it

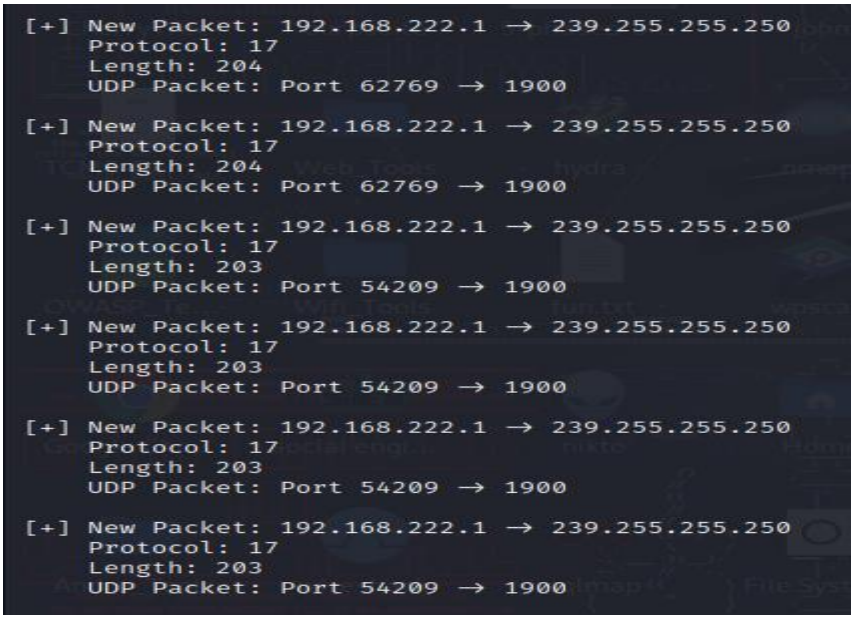


```
┌──(kali㉿kali)-[~]
└─$ sudo chmod +x network_sniffer.py
```

Now execute the file :



```
┌──(kali㉿kali)-[~]
└─$ sudo python3 network_sniffer.py

Starting network sniffer ...
[+] New Packet: 192.168.222.135 → 192.168.222.254
    Protocol: 17
    Length: 310
    UDP Packet: Port 68 → 67

[+] New Packet: 192.168.222.254 → 192.168.222.135
    Protocol: 17
    Length: 328
    UDP Packet: Port 67 → 68

[+] New Packet: 192.168.222.1 → 239.255.255.250
    Protocol: 17
    Length: 204
    UDP Packet: Port 62769 → 1900

[+] New Packet: 192.168.222.1 → 239.255.255.250
    Protocol: 17
    Length: 204
    UDP Packet: Port 62769 → 1900

[+] New Packet: 192.168.222.1 → 239.255.255.250
    Protocol: 17
    Length: 204
    UDP Packet: Port 62769 → 1900

[+] New Packet: 192.168.222.1 → 239.255.255.250
```



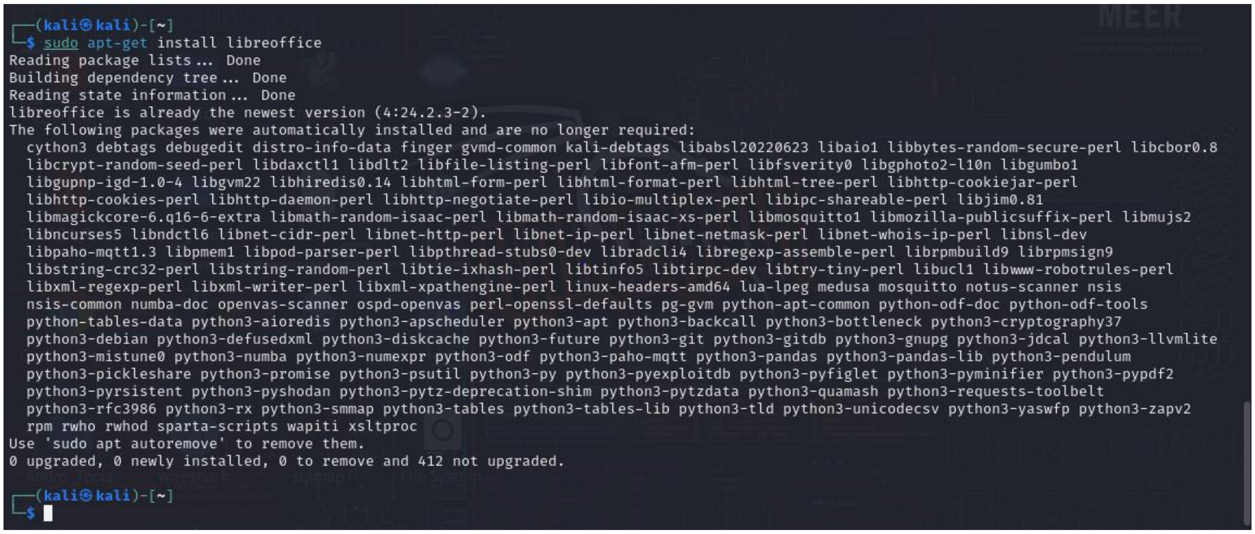
```
[+] New Packet: 192.168.222.1 → 239.255.255.250
    Protocol: 17
    Length: 204
    UDP Packet: Port 62769 → 1900

[+] New Packet: 192.168.222.1 → 239.255.255.250
    Protocol: 17
    Length: 204
    UDP Packet: Port 62769 → 1900

[+] New Packet: 192.168.222.1 → 239.255.255.250
    Protocol: 17
    Length: 203
    UDP Packet: Port 54209 → 1900

[+] New Packet: 192.168.222.1 → 239.255.255.250
    Protocol: 17
    Length: 203
    UDP Packet: Port 54209 → 1900

[+] New Packet: 192.168.222.1 → 239.255.255.250
    Protocol: 17
    Length: 203
    UDP Packet: Port 54209 → 1900

[+] New Packet: 192.168.222.1 → 239.255.255.250
    Protocol: 17
    Length: 203
    UDP Packet: Port 54209 → 1900
```

The script will start capturing packets and printing details like the source and destination IP addresses, the protocol used, and the length of each packet.

PHISHING AWARENESS TRAINING Create a presentation or online training module about phishing attacks. Educate others about recognizing and avoiding phishing emails, websites, and social engineering tactics

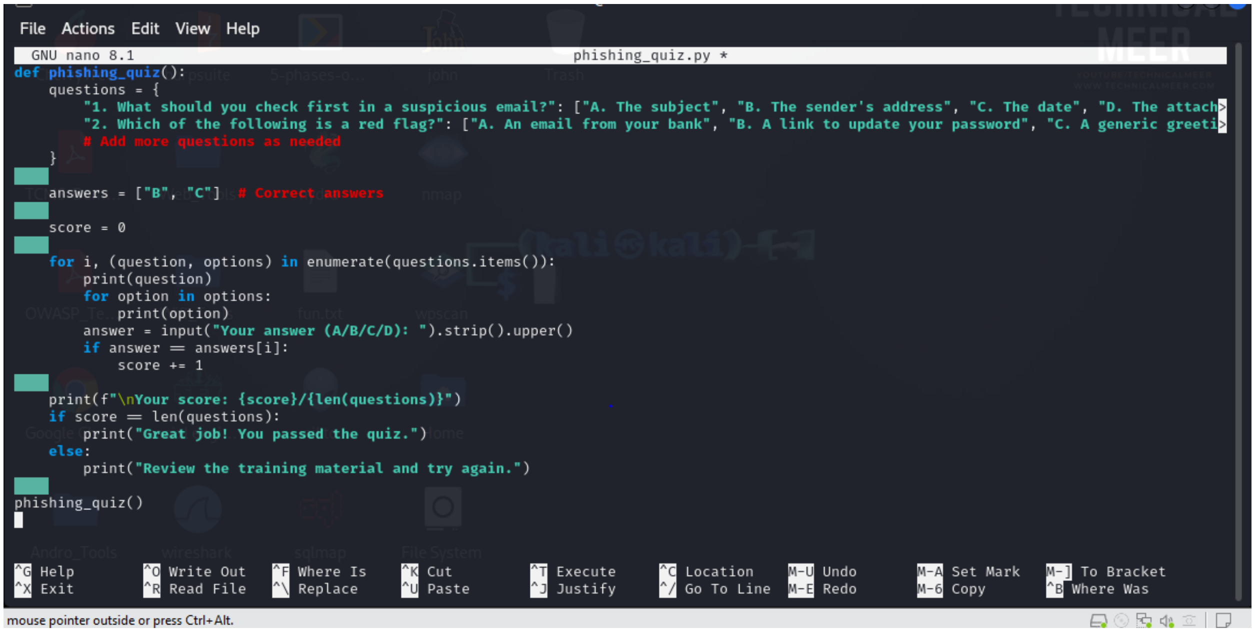
Answer

Installing the required libraries :



```
┌──(kali㉿kali)-[~]
└─$ sudo apt-get install libreoffice
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
libreoffice is already the newest version (4:24.2.3-2).
The following packages were automatically installed and are no longer required:
  cython3 debtags debugedit distro-info-data finger gvmd-common kali-debtags libabsl20220623 libaio1 libbytes-random-secure-perl libcbor0.8
  libcrypt-random-seed-perl libdaxctl1 libdlt2 libfile-listing-perl libfont-afm-perl libfsverity0 libgphoto2-l10n libgumbo1
  libgupnp-igd-1.0-4 libgvm22 libhiredis0.14 libhtml-form-perl libhtml-format-perl libhtml-tree-perl libhttp-cookiejar-perl
  libhttp-cookies-perl libhttp-daemon-perl libhttp-negotiate-perl libio-multiplex-perl libipc-shareable-perl libjim0.81
  libmagickcore-6.q16-6-extra libmath-random-isaac-perl libmath-random-isaac-xs-perl libmosquitto1 libmozilla-publicsuffix-perl libmujs2
  libncurses5 libndctl6 libnet-cidr-perl libnet-http-perl libnet-ip-perl libnet-netmask-perl libnet-whois-ip-perl libnsl-dev
  libpaho-mqtt1.3 libpmem1 libpod-parser-perl libpthread-stubs0-dev libradcli4 libregexp-assemble-perl librpmbuild9 librpmsign9
  libstring-crc32-perl libstring-random-perl libtie-ixhash-perl libtinfo5 libtirpc-dev libtry-tiny-perl libucl1 libwww-robotrules-perl
  libxml-regexp-perl libxml-writer-perl libxml-xpathengine-perl linux-headers-amd64 lua-lpeg medusa mosquitto notus-scanner nsis
  nsis-common numba-doc openvas-scanner ospd-openvas perl-openssl-defaults pg-gvm python-apt-common python-odf-doc python-odf-tools
  python-tables-data python3-aioredis python3-apscheduler python3-apt python3-backcall python3-bottleneck python3-cryptography37
  python3-debian python3-defusedxml python3-diskcache python3-future python3-git python3-gitdb python3-gnupg python3-jdcal python3-llvmlite
  python3-mistune0 python3-numba python3-numexpr python3-odf python3-paho-mqtt python3-pandas python3-pandas-lib python3-pendulum
  python3-pickleshare python3-promise python3-psutil python3-py python3-pyexploitdb python3-pyfiglet python3-pyminifier python3-pypdf2
  python3-pyrsistent python3-pyshodan python3-pytz-deprecation-shim python3-pytzdata python3-quamash python3-requests-toolbelt
  python3-rfc3986 python3-rx python3-smmap python3-tables python3-tables-lib python3-tld python3-unicodecsv python3-yaswfp python3-zapv2
  rpm rwho rwhod sparta-scripts wapiti xsltproc
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 412 not upgraded.

┌──(kali㉿kali)-[~]
└─$ 
```

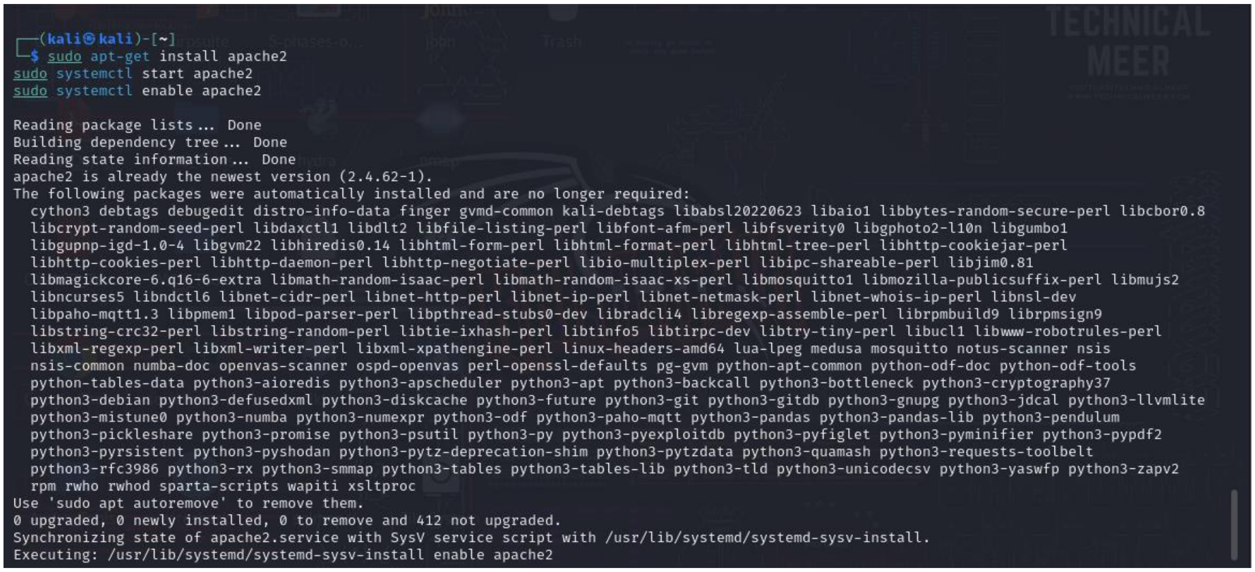
Creating a python file and coding it :



```
 GNU nano 8.1                                          phishing_quiz.py *
def phishing_quiz():
    questions = {
        "1. What should you check first in a suspicious email?": ["A. The subject", "B. The sender's address", "C. The date", "D. The attach>
        "2. Which of the following is a red flag?": ["A. An email from your bank", "B. A link to update your password", "C. A generic greeti>
        # Add more questions as needed
    }

    answers = ["B", "C"]  # Correct answers

    score = 0

    for i, (question, options) in enumerate(questions.items()):
        print(question)
        for option in options:
            print(option)
        answer = input("Your answer (A/B/C/D): ").strip().upper()
        if answer == answers[i]:
            score += 1

    print(f"\nYour score: {score}/{len(questions)}")
    if score == len(questions):
        print("Great job! You passed the quiz.")
    else:
        print("Review the training material and try again.")

phishing_quiz()
```

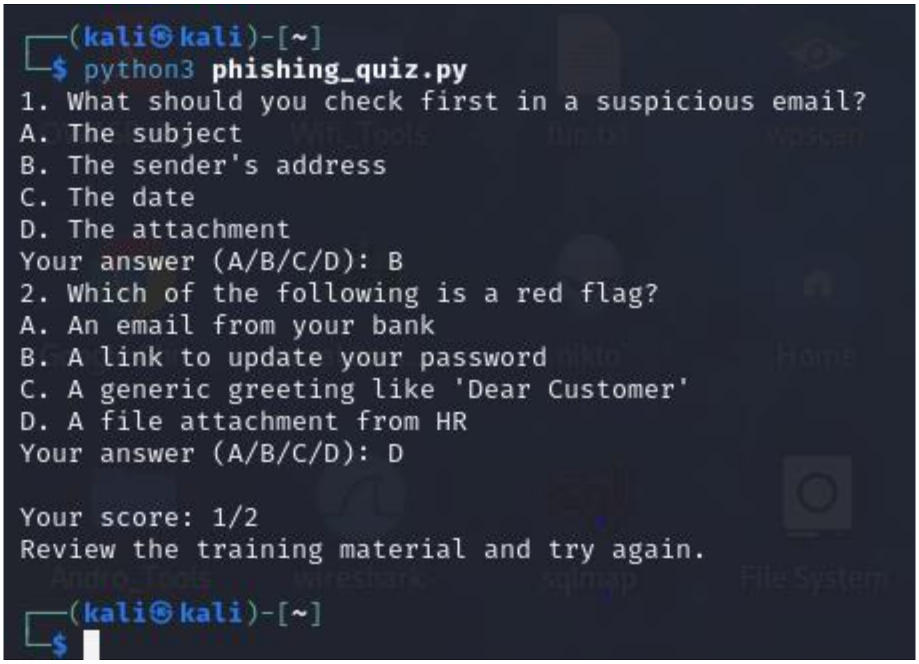
Now save it and execute it,

Now installing apache :



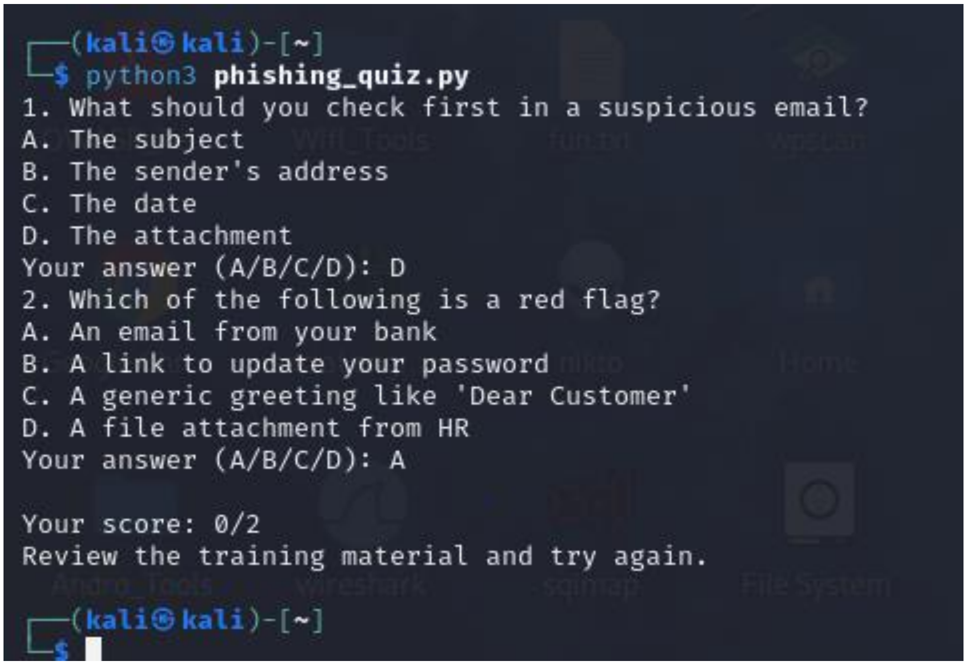
```
┌──(kali㉿kali)-[~]
└─$ sudo apt-get install apache2
sudo systemctl start apache2
sudo systemctl enable apache2

Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apache2 is already the newest version (2.4.62-1).
The following packages were automatically installed and are no longer required:
  cython3 debtags debugedit distro-info-data finger gvmd-common kali-debtags libabsl20220623 libaio1 libbytes-random-secure-perl libcbor0.8
  libcrypt-random-seed-perl libdaxctl1 libdlt2 libfile-listing-perl libfont-afm-perl libfsverity0 libgphoto2-l10n libgumbo1
  libgupnp-igd-1.0-4 libgvm22 libhiredis0.14 libhtml-form-perl libhtml-format-perl libhtml-tree-perl libhttp-cookiejar-perl
  libhttp-cookies-perl libhttp-daemon-perl libhttp-negotiate-perl libio-multiplex-perl libipc-shareable-perl libjim0.81
  libmagickcore-6.q16-6-extra libmath-random-isaac-perl libmath-random-isaac-xs-perl libmosquitto1 libmozilla-publicsuffix-perl libmujs2
  libncurses5 libndctl6 libnet-cidr-perl libnet-http-perl libnet-ip-perl libnet-netmask-perl libnet-whois-ip-perl libnsl-dev
  libpaho-mqtt1.3 libpmem1 libpod-parser-perl libpthread-stubs0-dev libradcli4 libregexp-assemble-perl librpmbuild9 librpmsign9
  libstring-crc32-perl libstring-random-perl libtie-ixhash-perl libtinfo5 libtirpc-dev libtry-tiny-perl libucl1 libwww-robotrules-perl
  libxml-regexp-perl libxml-writer-perl libxml-xpathengine-perl linux-headers-amd64 lua-lpeg medusa mosquitto notus-scanner nsis
  nsis-common numba-doc openvas-scanner ospd-openvas perl-openssl-defaults pg-gvm python-apt-common python-odf-doc python-odf-tools
  python-tables-data python3-aioredis python3-apscheduler python3-apt python3-backcall python3-bottleneck python3-cryptography37
  python3-debian python3-defusedxml python3-diskcache python3-future python3-git python3-gitdb python3-gnupg python3-jdcal python3-llvmlite
  python3-mistune0 python3-numba python3-numexpr python3-odf python3-paho-mqtt python3-pandas python3-pandas-lib python3-pendulum
  python3-pickleshare python3-promise python3-psutil python3-py python3-pyexploitdb python3-pyfiglet python3-pyminifier python3-pypdf2
  python3-pyrsistent python3-pyshodan python3-pytz-deprecation-shim python3-pytzdata python3-quamash python3-requests-toolbelt
  python3-rfc3986 python3-rx python3-smmap python3-tables python3-tables-lib python3-tld python3-unicodecsv python3-yaswfp python3-zapv2
  rpm rwho rwhod sparta-scripts wapiti xsltproc
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 412 not upgraded.
Synchronizing state of apache2.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable apache2
```

Output:



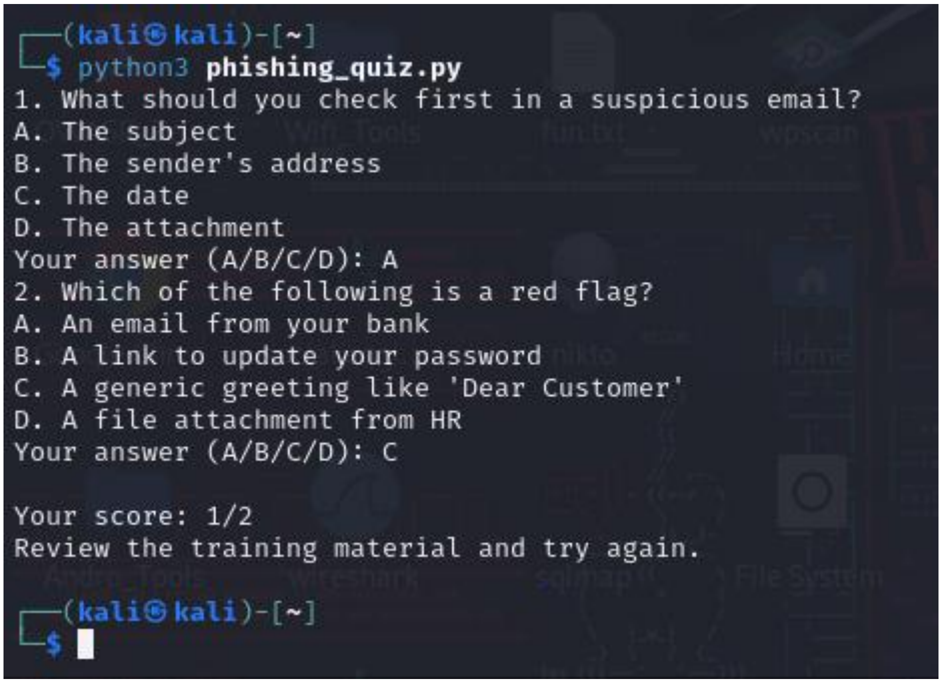
```
┌──(kali㉿kali)-[~]
└─$ python3 phishing_quiz.py
1. What should you check first in a suspicious email?
A. The subject
B. The sender's address
C. The date
D. The attachment
Your answer (A/B/C/D): B
2. Which of the following is a red flag?
A. An email from your bank
B. A link to update your password
C. A generic greeting like 'Dear Customer'
D. A file attachment from HR
Your answer (A/B/C/D): D

Your score: 1/2
Review the training material and try again.

┌──(kali㉿kali)-[~]
└─$ 
```

Output2:



```
┌──(kali㉿kali)-[~]
└─$ python3 phishing_quiz.py
1. What should you check first in a suspicious email?
A. The subject
B. The sender's address
C. The date
D. The attachment
Your answer (A/B/C/D): D
2. Which of the following is a red flag?
A. An email from your bank
B. A link to update your password
C. A generic greeting like 'Dear Customer'
D. A file attachment from HR
Your answer (A/B/C/D): A

Your score: 0/2
Review the training material and try again.

┌──(kali㉿kali)-[~]
└─$ 
```

Output3:



```
┌──(kali㉿kali)-[~]
└─$ python3 phishing_quiz.py
1. What should you check first in a suspicious email?
A. The subject
B. The sender's address
C. The date
D. The attachment
Your answer (A/B/C/D): A
2. Which of the following is a red flag?
A. An email from your bank
B. A link to update your password
C. A generic greeting like 'Dear Customer'
D. A file attachment from HR
Your answer (A/B/C/D): C

Your score: 1/2
Review the training material and try again.

┌──(kali㉿kali)-[~]
└─$ 
```

****