Disclaimer::-

To avoid legal issues I have permission to perform network enumeration on the target network

- The script combines different types of network enumerations to gather detailed information about the services running on the target machine.

Required Packages:

`nmap` (installed via terminal: `sudo apt-get install nmap`)

`impacket` (for NetBIOS: `pip install impacket`)

`ldap3` (for LDAP: `pip install ldap3`)

`pysnmp` (for SNMP: `pip install pysnmp`)

Code:-

```
import socket
import subprocess
from impacket.nmb import NetBIOS
from pysnmp.hlapi import *
from ldap3 import Server, Connection, ALL

# Function to scan for open ports on the target
def scan_ports(ip, ports):
    open_ports = []
    for port in ports:
        try:
            sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
            sock.settimeout(1)
            result = sock.connect_ex((ip, port))
            if result == 0:
                open_ports.append(port)
            sock.close()
        except socket.error:
            pass
    return open_ports

# Function to perform NetBIOS enumeration
def netbios_enumeration(ip):
    nb = NetBIOS(broadcast=False)
    try:
        node_status = nb.get_node_status(ip)
        if node_status:
            print(f"NetBIOS information for {ip}:")
            for entry in node_status:
```

```
                print(f"Name: {entry['NAME']}, Type: {entry['TYPE']}, Flags:
{entry['FLAGS']}")
        else:
            print(f"No NetBIOS information found for {ip}")
    except Exception as e:
        print(f"Error during NetBIOS enumeration: {e}")
    finally:
        nb.close()

# Function to perform SNMP enumeration
def snmp_enumeration(ip):
    try:
        iterator = getCmd(SnmpEngine(),
                          CommunityData('public'),
                          UdpTransportTarget((ip, 161)),
                          ContextData(),
                          ObjectType(ObjectIdentity('1.3.6.1.2.1.1.1.0')))
        error_indication, error_status, error_index, var_binds = next(iterator)

        if error_indication:
            print(f"SNMP Error: {error_indication}")
        elif error_status:
            print(f"SNMP Error: {error_status.prettyPrint()}")
        else:
            for var_bind in var_binds:
                print(f"SNMP Response: {var_bind}")
    except Exception as e:
        print(f"Error during SNMP enumeration: {e}")

# Function to perform LDAP enumeration
def ldap_enumeration(ip):
    try:
        server = Server(ip, get_info=ALL)
        conn = Connection(server)
        if conn.bind():
            print(f"LDAP Server Info: {server.info}")
        else:
            print(f"Failed to connect to LDAP server at {ip}")
    except Exception as e:
        print(f"Error during LDAP enumeration: {e}")

# Function to perform NTP enumeration
def ntp_enumeration(ip):
    try:
```

```
        result = subprocess.run(["ntpdate", "-q", ip], capture_output=True,
text=True)
        if result.returncode == 0:
            print(f"NTP Server Info for {ip}: {result.stdout}")
        else:
            print(f"Failed to get NTP info from {ip}")
    except Exception as e:
        print(f"Error during NTP enumeration: {e}")

# Function to run an Nmap scan
def nmap_scan(ip, ports):
    try:
        ports_str = ",".join(map(str, ports))
        result = subprocess.run(["nmap", "-sV", "-p", ports_str, ip],
capture_output=True, text=True)
        print(f"Nmap scan results for {ip}:\n{result.stdout}")
    except Exception as e:
        print(f"Error during Nmap scan: {e}")

# Menu-driven user interface
def show_menu():
    print("\nEnumeration Menu:")
    print("1. Scan Ports")
    print("2. NetBIOS Enumeration")
    print("3. SNMP Enumeration")
    print("4. LDAP Enumeration")
    print("5. NTP Enumeration")
    print("6. Run Nmap Scan")
    print("7. Exit")

def main():
    # Get the target IP from the user
    target_ip = input("Enter the target IP address: ")

    # Define the common ports for each service
    common_ports = [139, 445, 161, 389, 123]

    # Keep the menu active until the user exits
    while True:
        show_menu()
        choice = input("\nChoose an option (1-7): ")

        if choice == '1':
            # Scan for open ports
            open_ports = scan_ports(target_ip, common_ports)
```

```
            if open_ports:
                print(f"Open ports for {target_ip}: {open_ports}")
            else:
                print(f"No open ports found on {target_ip}")

        elif choice == '2':
            # NetBIOS Enumeration
            open_ports = scan_ports(target_ip, [139, 445])
            if 139 in open_ports or 445 in open_ports:
                netbios_enumeration(target_ip)
            else:
                print("NetBIOS ports are not open on the target.")

        elif choice == '3':
            # SNMP Enumeration
            open_ports = scan_ports(target_ip, [161])
            if 161 in open_ports:
                snmp_enumeration(target_ip)
            else:
                print("SNMP port (161) is not open on the target.")

        elif choice == '4':
            # LDAP Enumeration
            open_ports = scan_ports(target_ip, [389])
            if 389 in open_ports:
                ldap_enumeration(target_ip)
            else:
                print("LDAP port (389) is not open on the target.")

        elif choice == '5':
            # NTP Enumeration
            open_ports = scan_ports(target_ip, [123])
            if 123 in open_ports:
                ntp_enumeration(target_ip)
            else:
                print("NTP port (123) is not open on the target.")

        elif choice == '6':
            # Run Nmap scan on open ports
            open_ports = scan_ports(target_ip, common_ports)
            if open_ports:
                nmap_scan(target_ip, open_ports)
            else:
                print(f"No open ports found on {target_ip} for Nmap scanning.")
```

```
        elif choice == '7':
            print("Exiting the program...")
            break

        else:
            print("Invalid option, please choose again.")

# Run the menu-driven enumeration process
if __name__ == "__main__":
    main()
```

output:-

```
File  Actions  Edit  View  Help

┌──(root㉿kali)-[/home/kali/dharm/EHVA]
└─# python3 p4.py
Enter the target IP address: 192.168.16.122

Enumeration Menu:
1. Scan Ports
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP Enumeration
6. Run Nmap Scan
7. Exit

Choose an option (1-7): 1
Open ports for 192.168.16.122: [139, 445]
```

```
Enumeration Menu:
1. Scan Ports
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP Enumeration
6. Run Nmap Scan
7. Exit

Choose an option (1-7): 2
Traceback (most recent call last):
  File "/home/kali/dharm/EHVA/p4.py", line 170, in <module>
    main()
  File "/home/kali/dharm/EHVA/p4.py", line 125, in main
    netbios_enumeration(target_ip)
  File "/home/kali/dharm/EHVA/p4.py", line 24, in netbios_enumeration
    nb = NetBIOS(broadcast=False)
         ^^^^^^^^^^^^^^^^^^^^^^^^
TypeError: NetBIOS.__init__() got an unexpected keyword argument 'broadcast'
```

```
Enumeration Menu:
1. Scan Ports
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP Enumeration
6. Run Nmap Scan
7. Exit

Choose an option (1-7): 3
SNMP port (161) is not open on the target.
```

```
Enumeration Menu:
1. Scan Ports
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP Enumeration
6. Run Nmap Scan
7. Exit

Choose an option (1-7): 4
LDAP port (389) is not open on the target.
```

```
Enumeration Menu:
1. Scan Ports
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP Enumeration
6. Run Nmap Scan
7. Exit

Choose an option (1-7): 5
NTP port (123) is not open on the target.
```

```
Enumeration Menu:
1. Scan Ports
2. NetBIOS Enumeration
3. SNMP Enumeration
4. LDAP Enumeration
5. NTP Enumeration
6. Run Nmap Scan
7. Exit

Choose an option (1-7): 6
Nmap scan results for 192.168.16.122:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-17 13:42 EDT
Nmap scan report for 192.168.16.122 (192.168.16.122)
Host is up (0.00012s latency).

PORT    STATE SERVICE     VERSION
139/tcp open  netbios-ssn Samba smbd 4.6.2
445/tcp open  netbios-ssn Samba smbd 4.6.2

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.74 seconds
```