جامعة الشعب

قسم : هندسة الامن السيبراني

الاسم : اباذر محمد علي مزهر

الاسم : زينب عدنان حنون

شعبة / F

assigbment برمجة الامن السيبراني python

Certainly! Cybersecurity programming often involves creating scripts to automate

 security tasks, analyze data, or even test security measures. Below is an example

of a Python script that performs a basic cybersecurity task: scanning a network for

open ports using the `socket` library. This script can be a good starting point for a

cybersecurity assignment.

Network Port Scanner in Python

This script will scan a given IP address for open ports within a specified range.

```python
import socket
from datetime import datetime

# Function to scan ports
def port_scan(ip, port_start, port_end):
    print(f"Starting scan on host: {ip}")
    print(f"Time started: {datetime.now()}")

    open_ports = []

    for port in range(port_start, port_end + 1):
```

```python
        sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

        socket.setdefaulttimeout(1)


        result = sock.connect_ex((ip, port))


        if result == 0:

            open_ports.append(port)


        sock.close()


    print(f"Open ports on {ip}: {open_ports}")


    print(f"Time finished: {datetime.now()}")


# Main function
if __name__ == "__main__":

    target_ip = input("Enter the IP address to scan: ")

    start_port = int(input("Enter the start port number: "))

    end_port = int(input("Enter the end port number: "))


    port_scan(target_ip, start_port, end_port)
```

Explanation

1. **Importing Libraries:**

   - `socket` is used to create a connection and check if a port is open.

   - `datetime` is used to track the start and end times of the scan.

2. **Function `port_scan`:**

   - Takes an IP address, a start port, and an end port as input.

   - Iterates over the range of ports and checks if each port is open using `sock.connect_ex()`.

   - If the connection is successful (i.e., the port is open), it is added to the `open_ports` list.

   - Prints the list of open ports and the time taken for the scan.

3. **Main Function:**

   - Takes user input for the target IP address, start port, and end port.

   - Calls the `port_scan` function with the provided inputs.

Running the Script

1. Save the script in a file, e.g., `port_scanner.py`.

2. Run the script using a Python interpreter:

```sh
python port_scanner.py
```

3. Follow the prompts to enter the IP address and port range.

Notes

- Ensure you have permission to scan the target IP address. Unauthorized port scanning can be illegal and unethical.

- This is a basic example. In real-world applications, consider using advanced libraries such as `scapy` for network-related tasks.

- Additional features like multithreading can be added to speed up the scan.

This script can serve as a basic template for a cybersecurity programming assignment. You can extend it by adding features like scanning specific services, handling exceptions more gracefully, or integrating with other tools and libraries for enhanced functionality.