COMP 7005

Assignment 3

Design

Aditya Singh Attri

A01276335

Nov 9thth, 2024

# Purpose

# This program implements a TCP SYN-based port scanner that sends SYN packets to determine the status of specified ports on a target IP address. It helps identify open, closed, and filtered ports to assess network security and service accessibility.

# Data Types

**port\_scanner.py**

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| target\_ip | string | The IP address of the target machine to be scanned |
| start\_port | int | The starting port number for the scan |
| end\_port | int | The ending port number for the scan |
| delay | float | Delay between sending packets (in seconds) |
| port\_status | dict | Dictionary to store the status (open, closed, filtered) of each scanned port |

# Pseudocode

**scan\_port**

### Parameters

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Type** | **Description** |
| target\_ip | string | IP address of the target |
| port | int | Port number to scan |
| timeout | float | Time to wait for a response |

### Pseudocode

1. Craft a TCP SYN packet to the target IP and specified port.
2. Send the packet and wait for a response.
3. If a response is received:
   1. If the response is SYN-ACK, mark the port as Open.
   2. If the response is RST, mark the port as Closed.
4. If no response is received within the timeout, mark the port as **Filtered**.
5. Return the port status.

**main**

### Parameters

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Type** | **Description** |
| target\_ip | string | |  | | --- | | IP address of the target machine to be scanned |  |  | | --- | |  | |
| start\_port | int | |  | | --- | | Starting port number |  |  | | --- | |  | |
| end\_port | int | |  | | --- | | Ending port number |  |  | | --- | |  | |
| delay | float | Delay (in milliseconds) between port scans |

### Pseudocode

1. Parse command-line arguments for target IP, port range, and delay.
2. Validate input values for IP format, port range, and delay.
3. For each port in the specified range:
   1. Send a TCP SYN packet to the target on the current port.
   2. Wait for a response within the specified timeout.
   3. Determine the port status based on the response (SYN-ACK, RST, or no response).
   4. Store and print the status of each port.
4. Handle exceptions and ensure the program exits gracefully on errors.