**Project 4 Report: Port Scanner**

Prepared by: Chintan Shailesh Gosalia and Awani Marathe

B538: Computer Networks

## COMPONENTS OF THE PROJECT

The following are the major components of this project:

1. Command line parsing:

This component houses the code to parse the command line arguments given by the user. Here, depending on the arguments the user give, certain global data structures are populated. The values stored in these data structures are then used for further processing. It also is concerned with creation of jobs and stores these jobs in a queue which has a global scope.

2. Logic for implementing the scanning techniques:

This module consists of constructing appropriate packets depending on the scan techniques and sending them to the target port numbers at the IP given by the user. This module is also concerned with receiving the responses of the packets that are sent and interpreting the status of the port depending on the values of the in the fields of these packets. In addition to this, it also consists of functions that fetch the service names and service versions for certain ports.

3. Consolidating the results, concluding and printing the output:

This module is concerned with the consolidation of all the scan results of each of each Ip as a record. These records are then stored in a vector and are iterated individually to draw a conclusion. Further, this module also is concerned with giving a formatted output.

## CONCEPT USED FOR DRAWING CONCLUSIONS:

The idea of drawing a conclusion of the status of the port on the basis of the results returned by the scan techniques is centered around the following two points:

- The result returned by a scan technique is given more precedence over the other technique if the result of the former one is drawn on the basis of some response (packet) received and that of the other is drawn on the basis of certain assumptions (cases where we assume the result when no packets are received).
- Since TCP is more reliable than UDP, TCP scan techniques are given more priority over the UDP scan

technique while considering the scan results for drawing a conclusion, if responses are received for both the techniques.

- For example, in case:

In case of conflicting results, the following is done

SYN scan: Open

UDP scan: Closed

Since SYN is a TCP scan technique, priority is given to it over the UDP scan technique on the basis of reliability.

Further, consider one more example:

NULL scan: Open

ACK scan: Unfiltered

In this case, priority will be to ACK scan because a SYN/RST packet is received in response to ACK packet while implementing the ACK scan technique. On the other hand, NULL concludes a port is open only when it does not receive a response.

LIMITATIONS OF THE PROJECT CODE:

- Memory leaks in the project have not been addressed.
- When a large range of ports are scanned using multiple scan techniques and using multiple threads, the execution of the programs halts midway through (however, output for the scan ports is printed). We believe this is due to bad memory management in the project.
- DNS scanning for port 53 yields incorrect results.