**Abilities:**

* Traceroute between one or more job seekers and any other node in the network and find the nearest job seeker(s) to the target node.
* Spy on your neighbours, the job creator could direct the job seekers to report the IP address and MAC address for every live host who shares the same LAN with the job seeker, the job creator should detect the job seekers that share the same LAN if any.
* If the job fails (e.g. the job seeker or creator crash), you should be able to resume from the crash point and not start/restart the entire job.

**Deployment and Testing Instructions:**

\*A demo video is also included in the GitHub Repo if any confusion arises\*

1. \*Open Python Project in python ide (PyCharm)\*
2. \*Confirm all proper libraries have been installed\*
3. \*Run Server.py\*
4. \*Run Client.py\*
5. \*Run Client1.py\* (Client1.py is a carbon copy of Client.py, used to simulate running a standalone .exe)
6. (Client.py) Enter “LOGIN Steven 123 JobCreator”
7. (Client.py) Enter “VIEWJOBS”
8. (Client.py) Enter “CREATEJOB Steven PortDetection 10 1.1.1.92 25565”
9. (Client.py) Enter “VIEWJOBS”
10. (Client1.py) Enter “LOGIN Abdul 123 JobSeeker”
11. (Client1.py) Enter “VIEWJOBS”
12. (Client1.py) Enter “JOINJOB Steven PortDetection Abdul”
13. (Client.py) Enter “STARTJOB Steven PortDetection”
14. (Client1.py) Enter “VIEWJOBS”
15. (Client1.py) Enter “COMPLETEJOB Steven PortDetection 1.1.1.92 25565”
16. \*Check ClientOutput.txt for results\*

**Test Cases and Corresponding Output:**

Figure 1.0: IP Detection

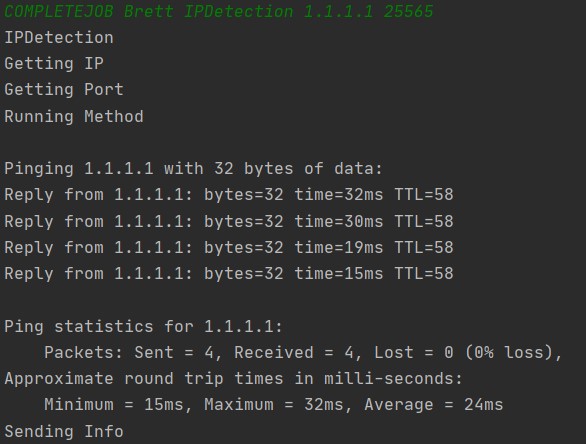


Figure 1.1:Port Detection

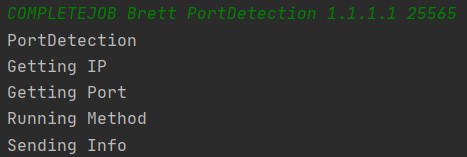


Figure 1.2: TCP Flood Attack

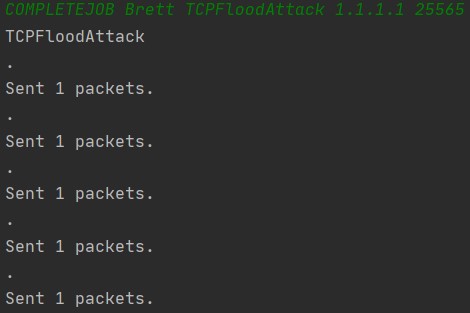


Figure 1.3: UDP Flood Attack

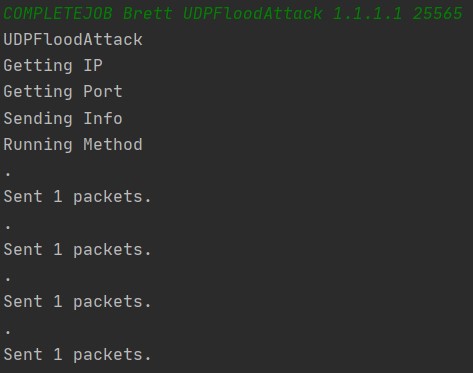


Figure 1.4: Node Location

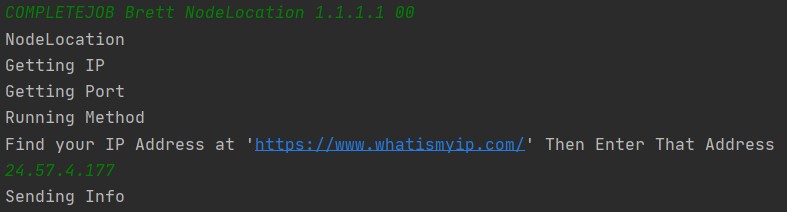


Figure 1.5: Node LAN Scan

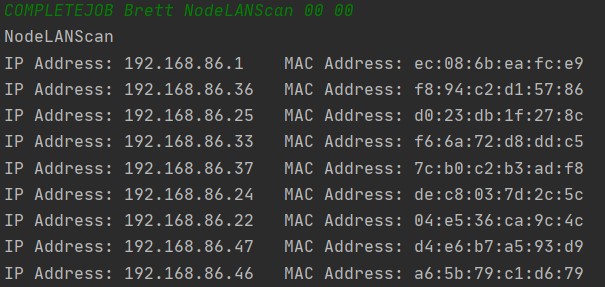


Figure 2.0: All Jobs Outputs

