Creating a Private Network Game Server

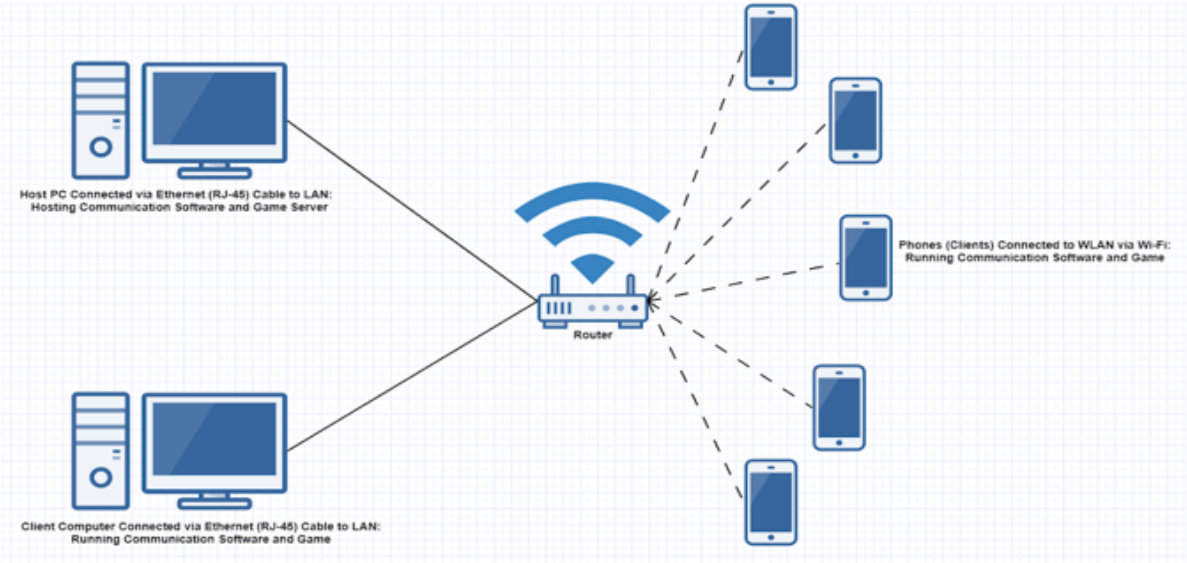

Figure 1 - Overall Diagram of Intranet Setup

Setting up a Game Server

After setting up the LAN communication software, I worked on creating a game server that allows cross-platform playing and offers a variety of administrative privileges. In order to accomplish this task, several LAN based cross-platform games were experimented with, including the likes of Chess, Scrabble, Connect Four, and Tic-Tac-Toe. However, none of these games offered the intuitive functionality that I desired. Hence, a different game, titled “Agar.io”, was used due to its cross-platform capability and use of administrative privileges. To make a private server for Agar.io, node.js and Microsoft Visual Studio Code (VSCode) were used. Despite the game’s usefulness, drawbacks in the form of poor administrative privileges and poor cross-platform capability were still present. Advanced features including banning players through IP, and playing with a large group of people were not possible. Also, only the scaled-down browser version worked for mobile devices. Due to Agar.io’s failings, the game “Minecraft” was used because of its renowned server capabilities. Banning players through their username and IP address, muting players, changing passwords, setting server capacity and instant messaging between players were things I could accomplish with Minecraft. The major hurdle I faced was the fact that Minecraft servers on Windows 7 run using Java, whereas all other devices use C++. In order to overcome this challenge, I researched different plugins on the internet that allowed me to use Minecraft Bedrock Edition Server Software on an unsupported OS. After finding a plugin, I used VSCode to edit the contents in the software and were able to run the server as shown in Figure 5. When it came time to run Minecraft on iOS and Android, I was faced with another difficult situation; the

fact that Minecraft now requires an Xbox Live sign-in, in order to play. This was not possible due to the lack of an internet connection in our private network. More research was done, and found that there are apps on both iOS (Minecraft Plug) and Android (BlockLauncher) that allowed me to bypass the login and connect to the server as shown in Figure 6. As a result, I was able to connect as many devices as I wanted. Overall, this project allowed me to gain many new skills about how to create and run a game server, and learn about the different games that allow cross-platform multiplayer over a LAN.

All in all, this project was a great learning experience. Learning about how to create and manage a LAN, and create game server has allowed me to gain valuable skills about both networking and about how to critically identify, analyze, and find solutions to problems.

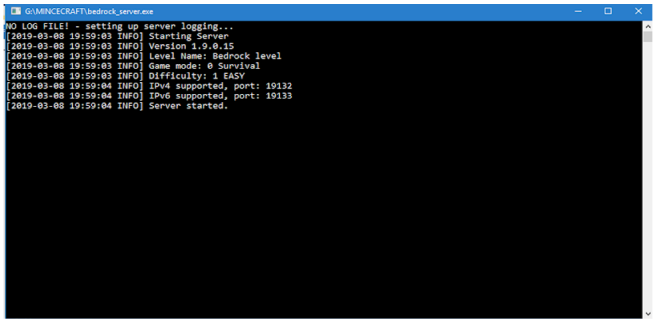

Figure 5 – Initializing Minecraft Server

Figure 6 - Minecraft Gameplay (Android)