University of Texas at Dallas

Smart Device Application Layer Protocol Project Report

Joah George

CS 4390 Computer Networks

Dr. Nhut Nguyen

August 3, 2023

Table of Contents

[Program Description 3](#_Toc141877581)

[Challenges 3](#_Toc141877582)

[Learned 4](#_Toc141877583)

[Algorithms and Techniques 4](#_Toc141877584)

[Suggestions 4](#_Toc141877585)

# Program Description

A brief explanation of the program from the user’s perspective is provided here. For information on the implementation of the Smart Device Protocol, see the SDP Implementation document.

When the client boots up, the user is presented with three operations to select from – PULL, PUSH, and EXIT – and a brief explanation of each. The user is then prompted to select an operation. Then, the device list will be displayed to the user which shows information about the available devices – their names, descriptions, and types. The user is prompted to enter the names of the devices that they would like to perform the operation previously selected on. The user can enter one or more names by putting a space between the names.

If the operation selected in the operation step was PUSH, then the client will display some guidance for values. The user is then prompted to enter a value to set the devices selected in the devices list step to. Then, the client will display the new device status(es) or that the device(s) could not be set to the value provided in the value step.

If the operation selected in the operation step was PUSH, then the client will display the current device status of each device provided in the device list step.

Then the operation list will appear again prompting the user for the next operation.

# Challenges

There were many challenges faced while making this project, but I will highlight the three that I believe were the toughest to overcome. The first major challenge was to establish the TCP connection between the client and server in Mininet. While I have done multiple projects previously that require a TCP connection, this was the first time I have used Python and Mininet. The primary difficulty was that I was not sure how to get the IP address of the hosts in Mininet using python. This was because when the gethostname function is used in Python, the hosts all shared the same hostname, and subsequently, they all shared the same IP address when the gethostbyname function is used on the shared hostname. For some reason, the hosts in Mininet were sharing a host name despite the ifconfig command showing different IP addresses. I eventually found an example of a Python TCP socket binding where the IP address field is left blank. I tried a blank IP address for the host and used the IP address in from the ifconfig in the client to connect to the host. This is the way that I was finally able to set up the TCP connection.

The second major challenge was designing a message that would fit the project requirements. The first message design was working wonderfully until I reread the project requirements and realized that there were multiple devices of a single device type (two light switches). While a redesign would have probably been the easiest solution, I didn’t want to completely scrap all the code I had already made with the first message design. I decided to modify the code to accommodate multiple devices with the same device type. This unnecessarily complicated the code, message creating, message parsing, and the server’s use of the message.

The third major challenge was some of the message were disappearing. When a user submitted a PULL or PUSH operation with multiple devices, the client would only get some of the messages. I checked the server, and the server was sending all the messages, but the client wasn’t receiving them. I noticed that when another PULL or PUSH operations was submitted, I would get the message of the previous operation. The messages weren’t completely disappearing but rather delayed. I can’t remember exactly how I found this out, but the cause was the server sending messages too fast to the client. I introduced a small amount of time between the servers messages and the messages were all displaying now.

# Learned

My primary takeaway from this project was that the TCP connection in socket programming is a powerful tool. This package allows programmers to very easily create top-level connections between servers and clients while also allowing anything to be send through the connection (once you figure out how to set it up properly). In Python, everything (objects, lists, dictionaries, etc.) can be easily converted to a JSON object and the JSON object can be easily converted to a string and sent through the TCP connection. Socket programming is a very powerful tool that reduces complexity of a program to enable faster and easier development of the functionality of the program rather than the underlying connections.

# Algorithms and Techniques

A technique I used to keep everything easy to understand was to always use a class for objects instead of a dictionary. This allowed my IDE to identify any syntax problems when I type the code instead of when I run the code. Classes also provide the ability to use inheritance. In the project, there are two devices – light switches and thermostats. With simple inheritance, I was able to store the fields that they shared in a single super class rather than on each of the subclasses. Additionally, the inheritance provided the base information needed to populate the device list.

A technique that I have used previously but did not use for this project was threading. Normally, when a server accepts a client, the server forks or threads so that it can continue to listen for more incoming clients. However, since there was only a single client that would be connecting, I decided to not use threading and keep the project simple.

# Suggestions

If the purpose of the project is to create the application layer protocol, then I believe that the students should focus on that. As mentioned before, I had trouble creating the TCP connection. Troubleshooting the TCP connection took up a lot of my time, but the purpose of the project was not to only set up the connection. I think that some skeleton code should be provided to the students so that they can focus on the application layer design rather than the TCP connection.