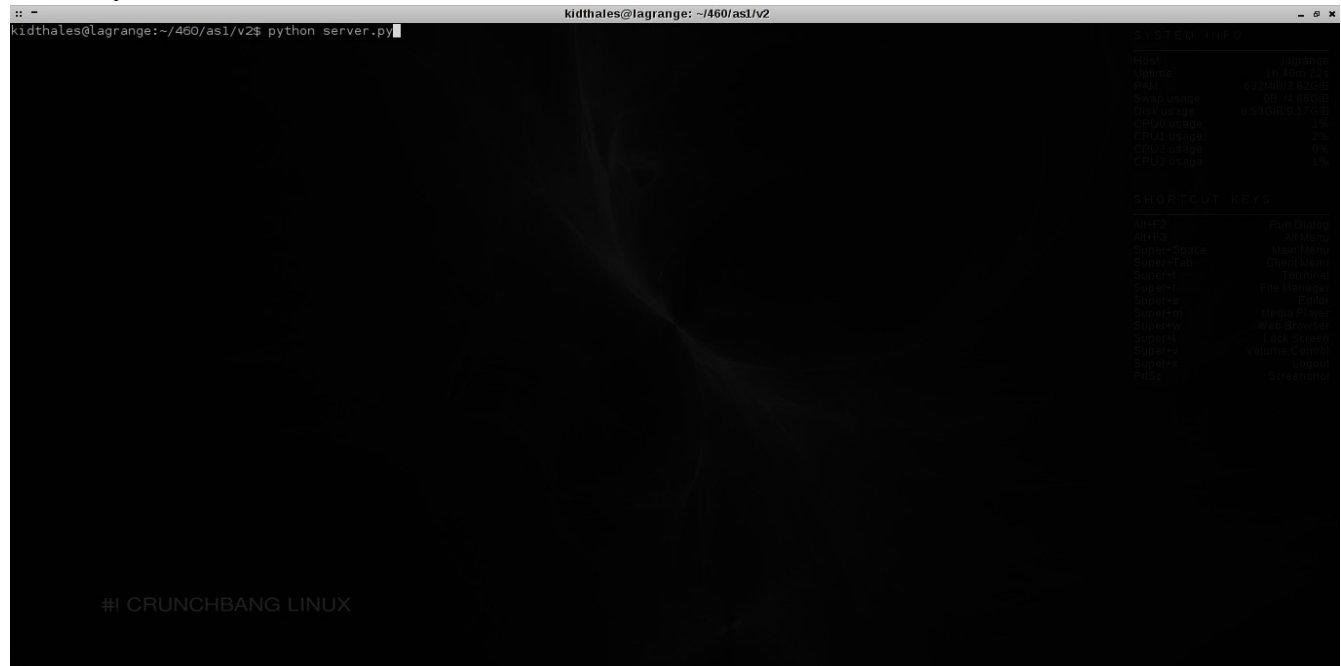


Server

Starting the Server

Navigate to the directory containing the file `server.py`. Type `python server.py` and press the enter key.



```
kidthales@lagrange: ~/460/as1/v2$ python server.py
```

SYSTEM INFO

Host	lagrange
Domain	lagrange
IP	127.0.0.1
OS	Ubuntu 20.04
Kernel	5.4.0-74-generic
Architecture	x86_64
CPU	Intel Core i7-9700K
GPU	NVIDIA GeForce RTX 2080
RAM	16GB
Storage	1TB

SHORTCUT KEYS

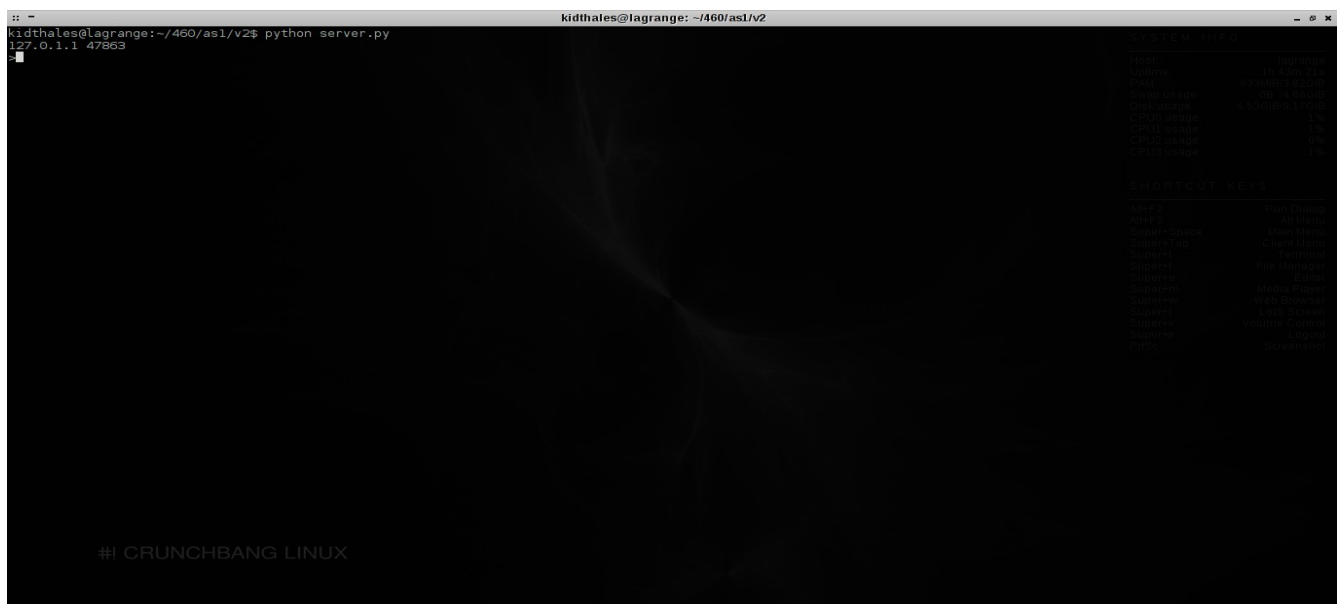
Ctrl+C	Quit Program
Ctrl+D	Quit Program
Ctrl+E	Quit Program
Ctrl+F	Quit Program
Ctrl+G	Quit Program
Ctrl+H	Quit Program
Ctrl+I	Quit Program
Ctrl+J	Quit Program
Ctrl+K	Quit Program
Ctrl+L	Quit Program
Ctrl+N	Quit Program
Ctrl+O	Quit Program
Ctrl+P	Quit Program
Ctrl+Q	Quit Program
Ctrl+R	Quit Program
Ctrl+S	Quit Program
Ctrl+T	Quit Program
Ctrl+U	Quit Program
Ctrl+V	Quit Program
Ctrl+W	Quit Program
Ctrl+X	Quit Program
Ctrl+Y	Quit Program
Ctrl+Z	Quit Program

#! CRUNCHBANG LINUX

Server Invocation

Server Operation

Upon start, the server will display the IP address and port number that the server is listening on for incoming connections. The server will also display a prompt for user input.



```
kidthales@lagrange: ~/460/as1/v2$ python server.py
127.0.0.1 47863
```

SYSTEM INFO

Host	lagrange
Domain	lagrange
IP	127.0.0.1
OS	Ubuntu 20.04
Kernel	5.4.0-74-generic
Architecture	x86_64
CPU	Intel Core i7-9700K
GPU	NVIDIA GeForce RTX 2080
RAM	16GB
Storage	1TB

SHORTCUT KEYS

Ctrl+C	Quit Program
Ctrl+D	Quit Program
Ctrl+E	Quit Program
Ctrl+F	Quit Program
Ctrl+G	Quit Program
Ctrl+H	Quit Program
Ctrl+I	Quit Program
Ctrl+J	Quit Program
Ctrl+K	Quit Program
Ctrl+L	Quit Program
Ctrl+N	Quit Program
Ctrl+O	Quit Program
Ctrl+P	Quit Program
Ctrl+Q	Quit Program
Ctrl+R	Quit Program
Ctrl+S	Quit Program
Ctrl+T	Quit Program
Ctrl+U	Quit Program
Ctrl+V	Quit Program
Ctrl+W	Quit Program
Ctrl+X	Quit Program
Ctrl+Y	Quit Program
Ctrl+Z	Quit Program

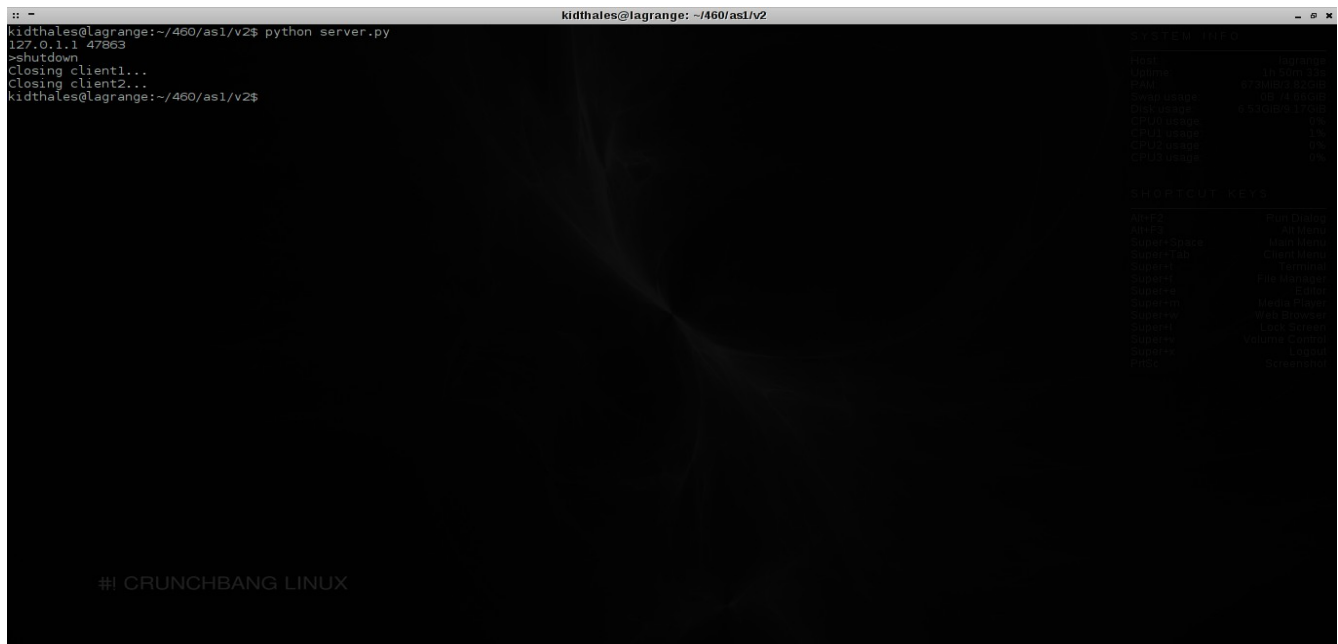
#! CRUNCHBANG LINUX

Server Operation

Server Shutdown

At the server prompt type **shutdown**. This will shutdown all clients currently connected to the server, as well as the server itself.

```
kidthales@lagrange:~/460/as1/v2$ python server.py
127.0.1.1 47863
>shutdown
Closing client1...
Closing client2...
kidthales@lagrange:~/460/as1/v2$
```



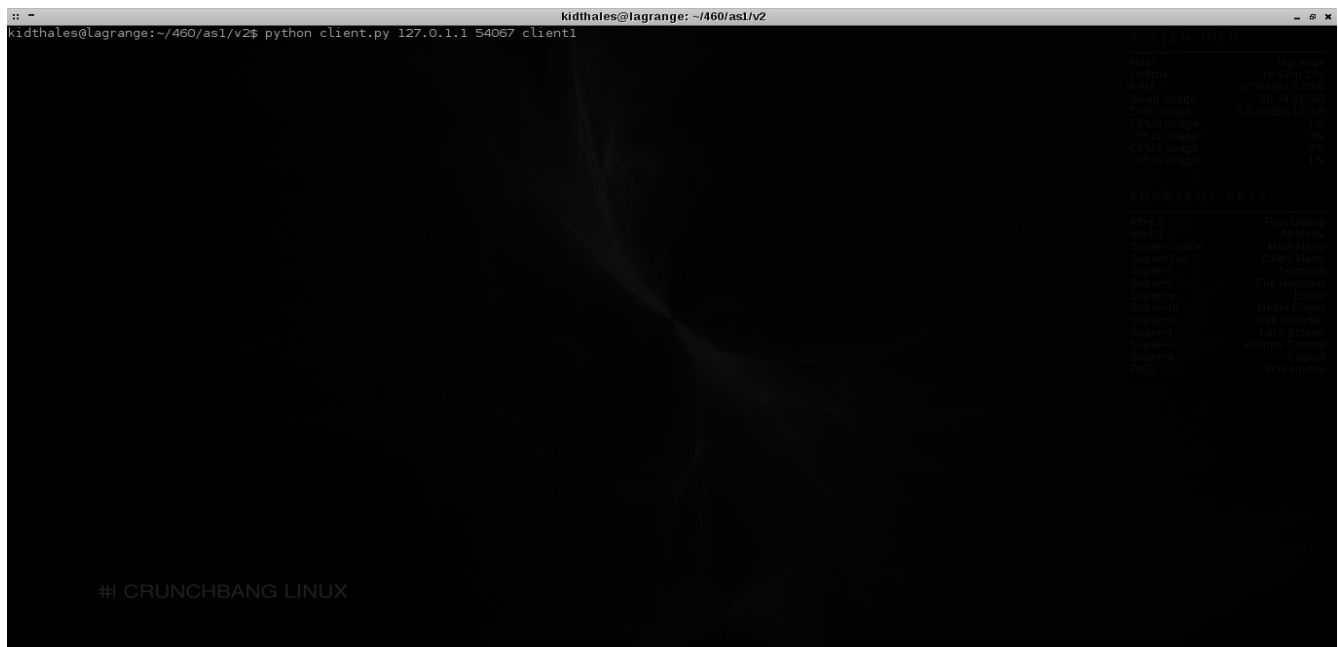
Server Shutdown - Notice all clients are closed remotely

Client

Starting a Client

Navigate to the directory containing the file **client.py**.

```
kidthales@lagrange:~/460/as1/v2$ python client.py 127.0.1.1 54067 client1
```

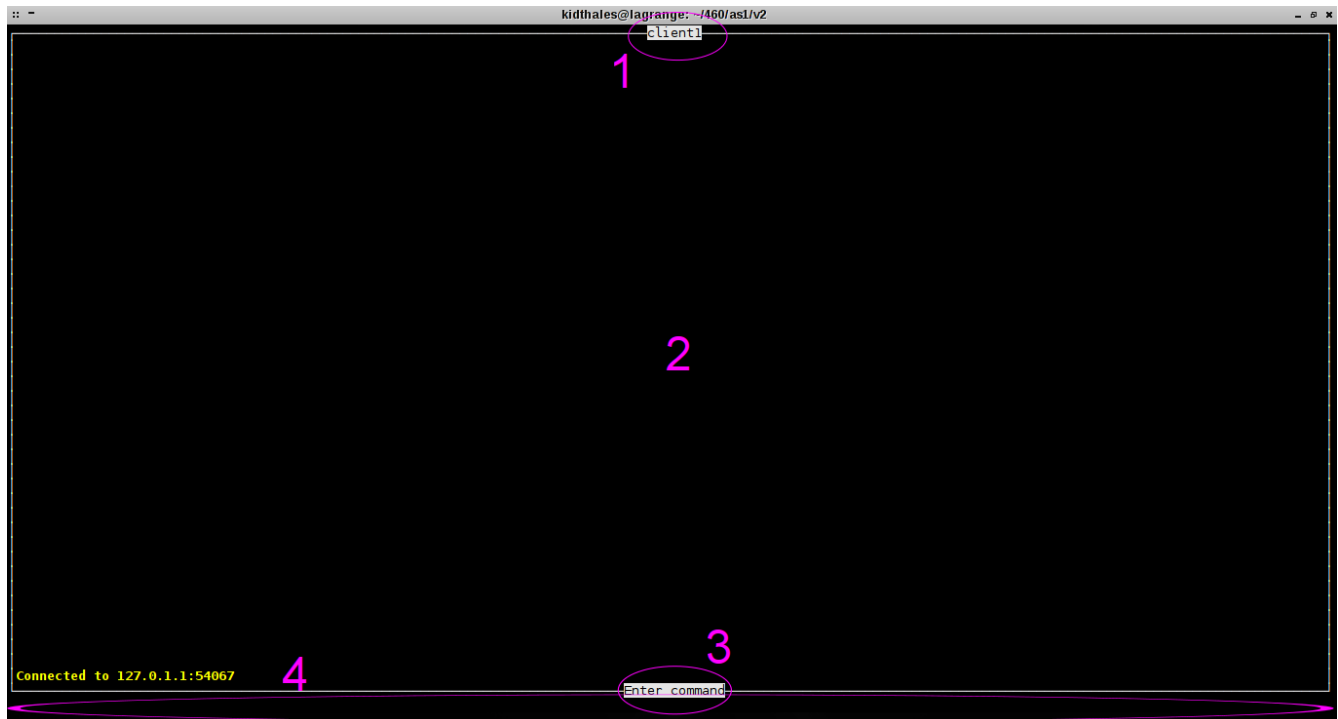


Client Invocation

Type `python client.py serverip serverport clientname` and press the enter key.

- `serverip` is the IP address that the server is listening on
- `serverport` is the port number the server is listening on
- `clientname` is the name you wish to register with the server; this name must be unique – it identifies your client with respect to other clients and the server

Client User Interface



Client User Interface

The user interface for the client is laid out as follows:

1. This label displays the name of the client
2. All messages received from clients is displayed in this area
3. This label provides context for user commands; multi-input commands will change this label based on the required input from the user
4. This field is for user input such as commands and messages to other clients

Client Operations Overview

There are three single input commands: `help`, `list`, and `quit`. There are two multi-input commands that correspond to message sending: `postcard` and `privatemessage`.

Client Operations – `help`

Provides a list and general overview of valid commands.

```
kidthales@lagrange: ~/460/as1/v2
client1

Connected to 127.0.1.1:54067
Enter command:
help
```

Client help Command - Input

```
kidthales@lagrange: ~/460/as1/v2
client1

Connected to 127.0.1.1:54067
: help
: Display help information.
: list
: List names of all clients connected to server.
: postcard
: Send a public message to one or multiple clients.
: privatemessage
: Send a private message to one client.
: quit
: De-register from server and terminate client.
Enter command:
```

Client help Command - Output

Client Operations – list

Lists names of all currently connected and registered clients. The querying client's name is also included in the list.

```
kidthales@lagrange: ~/460/as1/v2
client1

Connected to 127.0.1.1:54067
list
Enter command:
```

Client list Command - Input

```
kidthales@lagrange: ~/460/as1/v2
client1

Connected to 127.0.1.1:54067
client1 client2
Enter command:
```

Client list Command - Output

Client Operations – quit

De-registers client name with server and terminates client execution.

```
kidthales@lagrange: ~/460/as1/v2
client1

Connected to 127.0.1.1:54067
quit
Enter command
```

Client quit Command - Input

```
kidthales@lagrange: ~/460/as1/v2
client1

Connected to 127.0.1.1:54067
Client initiated shutdown...
Enter command
```

Client quit Command - Output

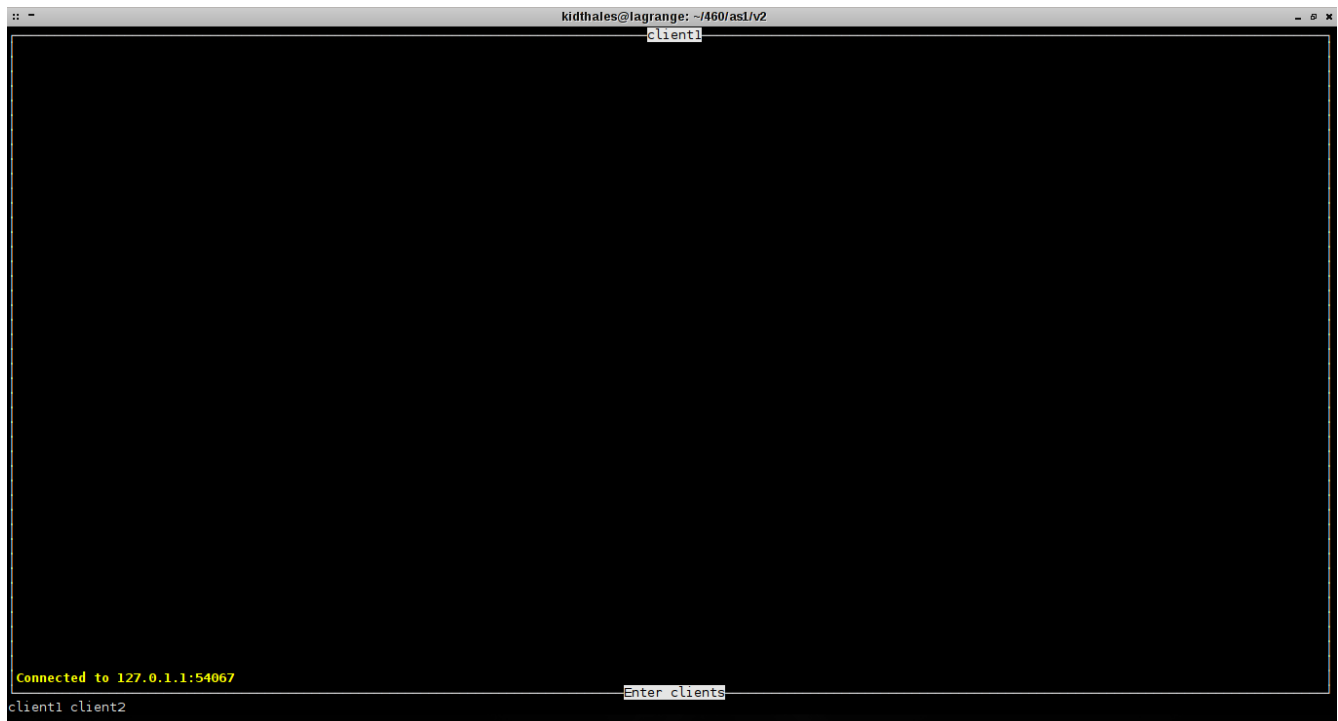
Client Operations – postcard

Upon entering this command, the user will be prompted for a list of clients (delineated by whitespace) to send a message to. Upon entering the clients, the user will be prompted to enter a one line message. This message will be sent to all clients in the given list that are registered with the

server; output of the entire command is also displayed in the sending client.



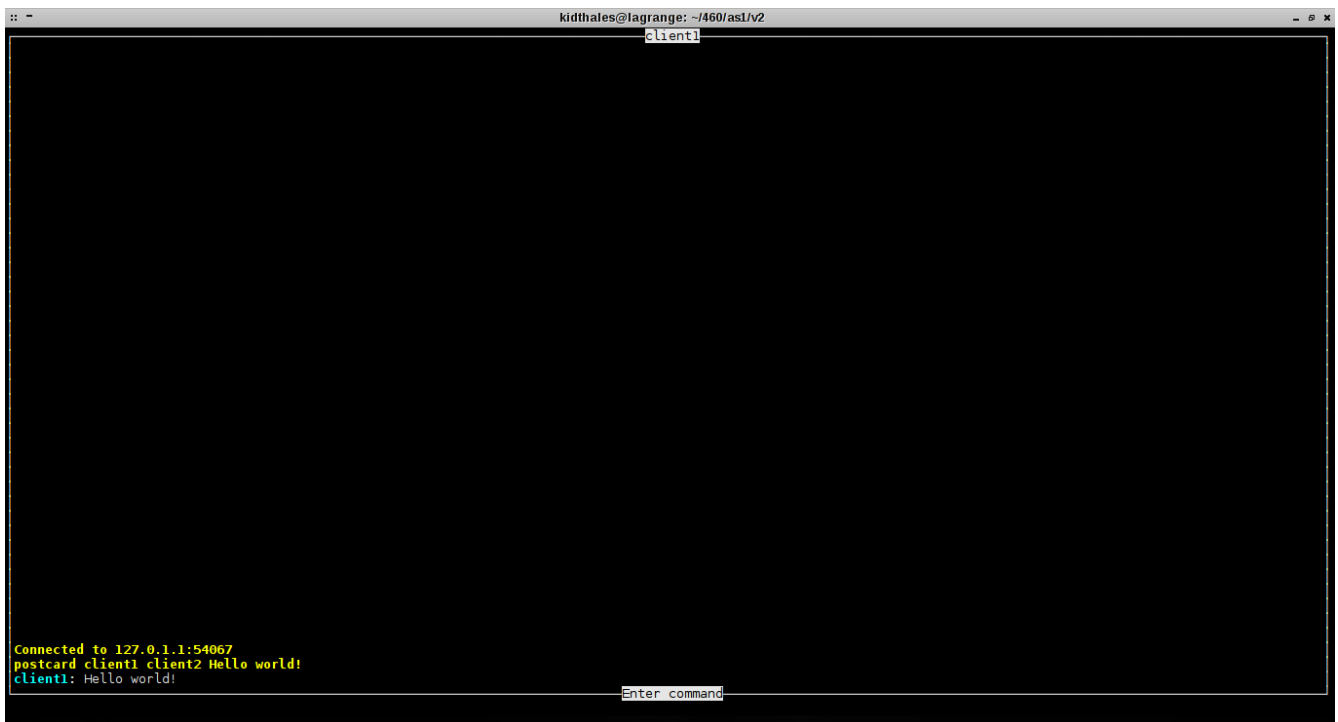
Client postcard Command - Input



Client postcard Command - Client List Input (Note the command context label)



Client postcard Command - Message Input (Note the command context label)



Client postcard Command - Sender Output (In this example the sender sent the message to himself and one other)


```
kidthales@lagrange: ~/460/as1/v2
client2

Connected to 127.0.1.1:54067
client1: Hello world!
Enter command
```

Client postcard Command - Receiver Output

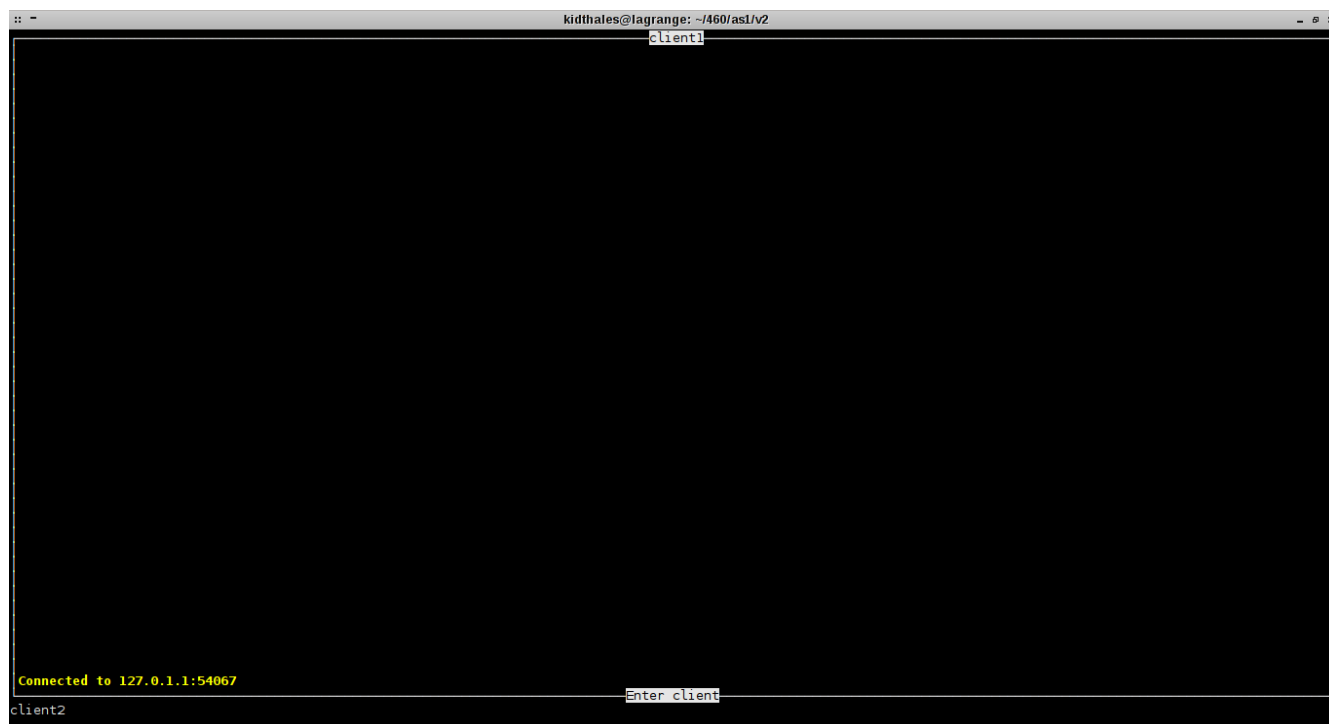
Client Operations – **privatemessage**

Upon entering this command, the user will be prompted for a client to send a private message to. Upon entering the client, the user will be prompted to enter a one line message. This message will be sent to the chosen client without passing through the server; output of the entire command is also displayed in the sending client.

```
kidthales@lagrange: ~/460/as1/v2
client1

Connected to 127.0.1.1:54067
privatemessage
Enter command
```

Client privatemessage Command - Input



Client privatemessage Command - Client Input (Note the command context label)



Client privatemessage Command - Message Input (Note the command context label)

```
kidthales@lagrange: ~/460/as1/v2
client1

Connected to 127.0.1.1:54067
privatemessage client2 Commence operation overlord.
Enter command
```

Client privatemessage Command - Sender Output

```
kidthales@lagrange: ~/460/as1/v2
client2

Connected to 127.0.1.1:54067
client1: Hello world!
client1: Commence operation overlord.
Enter command
```

Client privatemessage Command - Receiver Output