Chat (Timing) Covert Channel

Write a Python program that can extract a covert message from the delays in between characters of a message transmitted by a chat server. This is a group programming assignment (i.e., **each group will submit one program**). Please make sure that only one member of your group submits the program!

Notes and Requirements:

- Submit your source code only;
- Your program should connect to the chat server, receive an overt message, and display it as it is being received (to `stdout`);
- Your program should accurately time the delays between characters received of an overt message;
- Your program should determine the end of the overt message to properly disconnect from the chat server (hint: the overt message ends with "EOF"); and
- Your program should output the covert message (to `stdout` – hint: it also ends with "EOF").

Please, no GUIs. Make this a command line application without frills that I can execute at the command line as illustrated below:

```
jgourd@jgourd-latech:~/covert$ python chat_client.py
[connect to the chat server]
...
Some sort of overt message is being transmitted here.  But there is a
hidden message being covertly transmitted!  Can you guess it?
...
[disconnect from the chat server]
Covert message: Gourd is 31337!  Of course, we already knew this...
```

How can the delays that map to 0 or 1 be determined? I suggest that the timing between characters of the overt message be displayed to `stdout` – using pretty good precision (e.g., three decimal places). You should be able to pick out the two delays. Modify your program so that one delay maps to 0 and the other to 1. Hopefully, you guessed correctly (you have a 50% chance, after all). Of course, if that doesn't work, then switch the two! If you find that you are getting many different delays, it's likely that you are doing this on Windows or on a VM within a Windows host. We think that there is a bug in the Python timing library on Windows. Go figure...