

### Milestone 3 Project Report

In Milestone3, we defined a struct named packet, the packet for message between cproxy and sproxy. The packet contains type, payload, ack and length. For type, we used integer 1, 2, 3, to express heartbeat, message and session initiation. From Milestone 2 we know Cproxy can receive message from telnet or Sproxy. And Sproxy can receive message from server daemon or Cproxy.

In this project, instead of sending string message, we let Cproxy and Sproxy sending struct. When Cproxy receive string message from telnet, we will set the type as 2 and set the message length, then forward this struct to Sproxy. When Cproxy received packet from Sproxy Cproxy will detect the packet type, if it is 1 means Cproxy received ACK from Sproxy, if it's message we will forward to telnet. The Sproxy implementation is similar to Cproxy, when Sproxy received from Cproxy, it will see the packet type, if the type is 1, then Sproxy will send ACK to Cproxy, if the type is 2, then it will forward to server daemon. If the Sproxy received from server daemon, then it will set type as 2 then forward to Cproxy.

We use select function to send heartbeat every seconds. Three seconds not receiving anything, Sproxy will close the Cproxy socket, create a newsocket and waiting for accept Cproxy, Cproxy will close the Sproxy socket, create a newsocket and connect to Sproxy.

When user typed "logout", Sproxy will detect the recv() error from telnet daemon and then send a message with type 3 to Cproxy and reconnect with telnet daemon. Cproxy will receive a message with type 3 from sproxy. Then Cproxy will close the connection with telnet and create a new socket waiting for new telnet to connect. When user type "telnet localhost 5200", all the process will work again. During this disconnection, Cproxy and Sproxy should not end and keep running.