**Exception Handling - Design Review**

**Team2:** Hardik Barot & Sushant Deshpande

When incorrect API calls are made:

1. **bind() on _LISTEN:**
   - We don’t need to handle this case as it is taken care of by inet_bind().
   - It will return a -EINVAL.
     ```
     [EINVAL] socket is already bound to an address and the protocol
does not support binding to a new address.
     ```

2. **connect() on _SYN_SENT:**
   - We handle this case by adding the following code to the connect function:
     ```c
     if(sk->sk_state==TCP_SYNSENT)return -EALREADY
     ```
     ```
     [EALREADY] A connection request is already in progress for the specified socket.
     ```

3. **listen() on _ESTABLISHED:**
   - We handle this case by adding the following code to the listen function:
     ```c
     if(sk->sk_state==TCP_ESTABLISHED)return -EISCONN
     ```
     ```
     [EISCONN] Transport Endpoint already established.
     ```

When incorrect or unexpected packet types are received:

1. **syn1 on _CLOSE:**
   - The cop_rcv() function will take care of this by dropping the packet.

2. **syn2 on _LISTEN:**
   - Send back a CPF_SUPV message by setting the corresponding RESET bits in flags.

3. **data on _SYN_SENT:**
   - Drop the packet & send SYN1 again.