# NAL FAQ

1. Nal\_bind()🡪 一定要給IP address，若填”IP\_ADDR\_ANY”會由系統選route的IP address當作local address，但route是一對多，這樣的話收進來的packet可能會被分到別的device中。
2. 關於Nal\_connect()的return value之疑問：

當connect()在最後成功時會將dest addr & port的資訊塞進做connect()時使用的socket descriptor的結構中並返回該socket descriptor，因此文件上的return value是對的。

1. Nal\_send()/nal\_recv()時，發生0 byte時的情況為何?

回傳值>0代表成功傳送的值，若error 發生時會有相對應的error code，這邊有看一下底層的實作，在最底層的send data中若回傳0 byte的話他會依據socket性質回傳NAL\_WOULD\_BLOCK或其他error code，因此以nal\_send()/nal\_recv()這一層來說，不會有等於0的值回傳，因此上層使用者不需要處理0 byte的case。

1. Nal\_sendmsg()/ nal\_recvmsg() 和ㄧ般的nal\_send()/nal\_send\_to()/nal\_recv()/nal\_recv\_from()的主要差別🡪 透過nal\_msghdr傳資料，此structure中有含I/O vector struct來支持不連續的讀寫。
2. Nal\_fd\_reset() : 1🡪0 nal\_fd\_set(): 0🡪1
3. Nal\_shutdown() & nal\_close\_socket()的差別：nal\_shutdown可以控制要結束單邊或雙邊的連線，但socket descriptor並未釋放掉，例如：控制此socket不再receive data 或是控制此socket不再write data給某個socket。

而nal\_close\_socket()則是真的將socket釋放掉，結束整個connection。

1. Non-blocking socket下NAL\_CONNECT需搭配NAL\_IS\_CONNECTED來確保連線。
2. 延續7.，若不久後server side斷線了，使用NAL\_IS\_CONNECTED是否還會是連線狀態呢? 否 (底層的protocol 會幫忙改status，不好意思之前跟您講的有誤)

# NAL Interface & Filter 需注意事項

1. Nal\_bind()之後才能夠使用nal\_add\_filter()
2. Nal\_add\_filter()要在nal\_connect() or nal\_send\_to()之前使用，因為一設filter後若connect上了，對方可能就馬上會送東西過來，若nal\_add\_filter()在之後設，可能會有掉data的風險。
3. Nal\_add\_filter()若都沒有設就去nal\_connect() or nal\_send\_to()不會報錯，但會有warning，提醒使用者因為filter沒設，所以不會有data進來。
4. NAL\_SO\_BINDTODEVICE 必須要設在nal\_connect() or send/receive data之前。此socket option是用來bind socket到某特定的device，這樣收送資料才會從同一條path進出。
5. 設定NAL\_SO\_LINGER可以避免close socket的時候，因底層還沒真的release socket descriptor 而上層新創socket時又去用了同樣的socket descriptor。