Command:

* **Version**
  + Dùng để yêu cầu metadata blockchain
  + Gửi: h của blockchain, IP máy.
  + Nhận: so sánh h, nếu h > h local thì sẽ gửi **getblocks**, ngược lại gửi **version**
* **Getblocks**
  + Dùng để yêu cầu danh sách block (mã hash)
  + Nhận: gửi thông tin blockchain (mã hash các block) cho bên gửi bằng command **inv**
* **Inv**
  + Dùng để show cho peer khác danh sách block hoặc transaction mà máy mình có.
  + Gửi: có 2 kiểu là **tx** và **block,** danh sách block/transaction
  + Nhận: So sánh với dữ liệu hiện có, nếu thiếu thì dùng command **getdata** để yêu cầu thông tin chi tiết.
* **Getdata**
  + Dùng để yêu cầu thông tin về 1 transaction/block
  + Được dùng sau khi peer nhận được **inv**
  + Phản hồi bằng **block** hoặc **tx**
* **Block**
  + Gửi thông tin 1 hoặc nhiều block cụ thể
* **Tx**
  + Gửi thông tin 1 hoặc nhiều tx cụ thể
  + Gửi (thường là wallet node): gửi thông tin tx
  + Nhận:
    - Central node: Gửi **inv** về tx cho các node khác.
    - Các node khác: gửi lại getdata

Khi node được start:

* Nếu không phải central node
  + Gửi command **version** cho central node
  + Central phản hồi bằng **version**
    - Node gửi **getblocks**
    - Central phản hồi bằng **inv**
    - Node so sánh các hash block nhận được với dữ liệu của nó
    - Node gửi **getdata**
    - Central phản hồi bằng **block**
  + Central phản hồi bằng **getblocks**
    - Node gửi **inv**
    - Central so sánh, gửi **getdata**
    - Node phản hồi bằng **block**
* Lắng nghe command

Khi wallet node tạo transaction:

* Wallet node gửi **tx** cho central
* Central gửi **inv** tx cho các node còn lại
* Miner nhận tx, gửi **getdata** tx cho central
* Miner thấy pool có đủ 2 transaction, tiến hành đào.
* Miner gửi **inv** cho các node còn lại
* Các node gửi **getdata** block cho miner
* Miner gửi **block**
* Các node thêm block vào blockchain local của mình