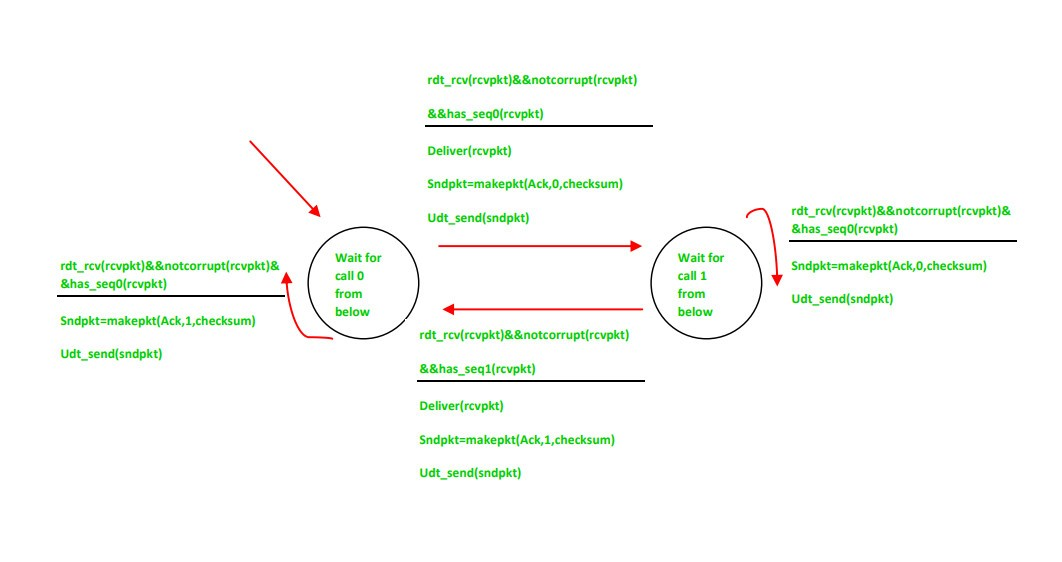
**Câu 4. Vì sao phải dùng các nguyên lý truyền tin tin cậy? So sánh sự khác nhau của các nguyên lý rdt 1.0, rdt 2.0, rdt 2.1 ,2.2 và rdt 3.0.**

Các nguyên lý truyền tin tin cậy (reliable data transfer - RDT) được áp dụng trong việc truyền dữ liệu qua mạng để **đảm bảo tính tin cậy** và **đúng đắn** của thông tin. Việc sử dụng các nguyên lý RDT quan trọng vì nó giúp giải quyết các vấn đề có thể xảy ra trong quá trình truyền dữ liệu, như mất mát dữ liệu, lỗi bit, sự trễ trong việc truyền tải, và thứ tự không đúng của các gói tin.

So sánh các nguyên lý truyền tải:

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Đặc điểm | Bên gửi | Bên nhận |
| rtd 1.0 | Truyền dữ liệu tin  cậy. (Giả sử  không có lỗi hay mất mát gì) | Gửi gói tin | Nhận gói tin |
| rtd 2.0 | Kênh truyền không làm mất gói. (Nhưng gói tin truyền có thể bị sai sót) | Gửi gói, đợi phản hồi (Stop and Wait Protocol) | Dùng checksum, ACK (Acknowledgement) và NAK (Negative ACK) để check lỗi gói tin gửi về sender |
| rtd 2.1 | Kênh truyền có ACK, NAK bị lỗi (ACK, NAK bị sai checksum) | Đánh số thứ tự 0, 1 cho các gói từ sender, gửi lại gói nếu NAK hoặc ACK/NAK bị lỗi (nhận biết bằng checksum trong gói), nhận được ACK thì mới gởi gói đánh số tiếp theo | Có cơ chế loại packet bị trùng |
| rtd 2.2 | Không dùng NAK, thay bằng gửi ACK gói gần nhất nhận thành công | Gửi lại gói nếu nhận ACK trùng lặp | Gửi ACK gói gần nhất thành công, đánh số thứ tự 0,1 cho ACK |
| rtd 3.0  => Hiệu suất  thấp:  Usender =  (L/R)/(RTT+L/R)  do phải nhận ACK rồi mới gửi  gói tiếp được | Xuất hiện mất gói | Chờ ACK trong khoảng thời gian “Hợp lí”. Dùng Timeout (Bộ định thì). Gửi lại pckt khi hết thời gian. - Các trường hợp gởi lại gói: Mất gói, mất ACK, thời gian chờ ngắn/ delayed ACK |  |

**5. Mô hình FSM của rdt 3.0 bên nhận:**



Reliable Data Transfer (3.0): Receiver FSM

Tài liệu tham khảo:

https://www.geeksforgeeks.org/reliable-data-transfer-rdt-3-0/