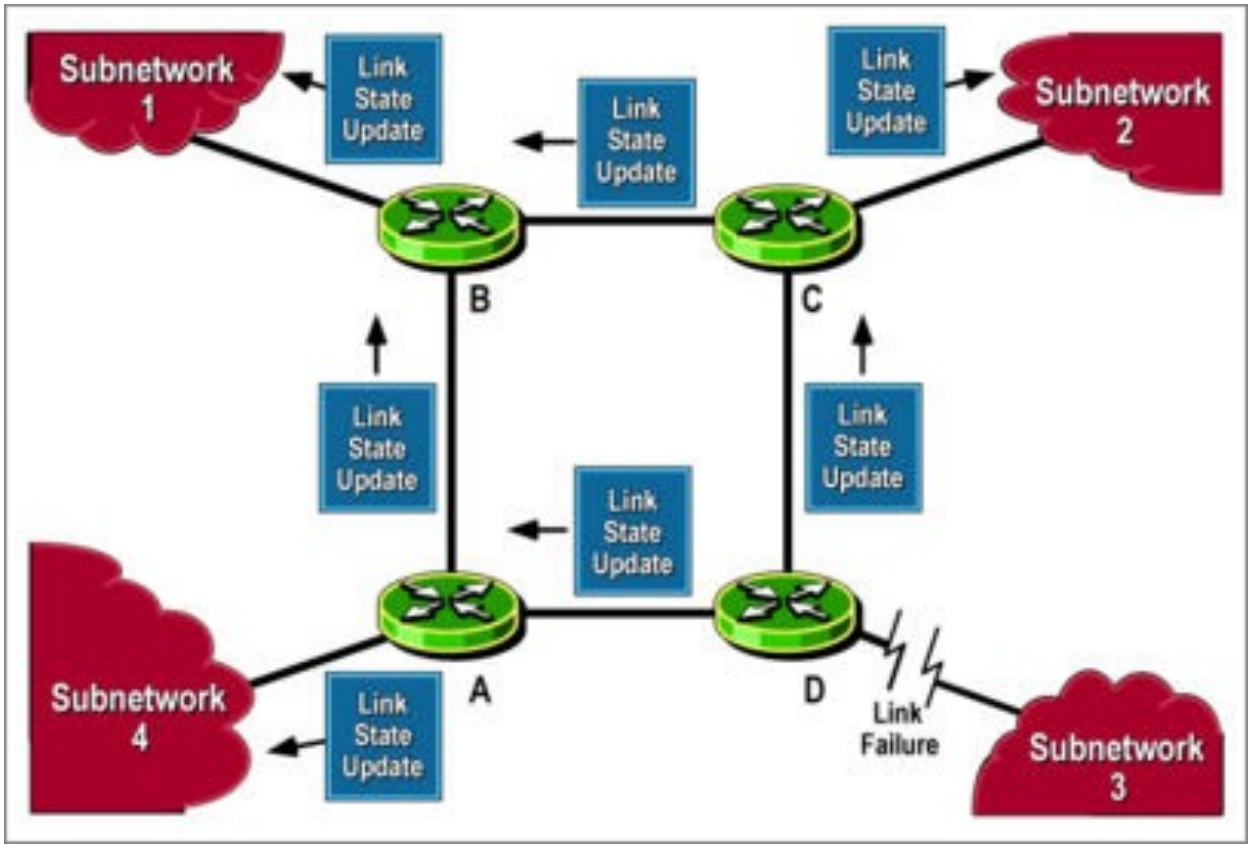
Distance Vector Routing Protocol



Team Members

Brandon Lawrence, Miguel Obiang,
Francisco Martínez, Adetomilola Popoola,
Reynaldo Gonzales

Team NO.403

Fall 2015

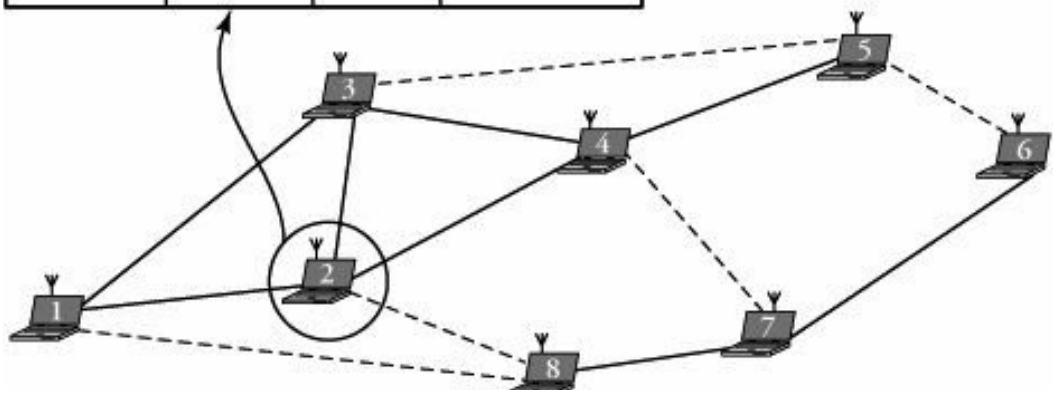
Distance Vector Routing Protocol

INTRODUCTION:

The objective of this project was to develop an application that emulates the distance vector routing protocol network. Simulating how a DVR works, we had each routing node periodically exchange routing updates with its neighbors, containing each node's entire routing table. Then upon receiving a routing update, a node shall update its routing table with the best routs to each destination; yet, each individual node shall have to remove the entries from its routing table if they have not been reconnected in a long period of time.

Routing Table for Node 2

Destination	Next Hop	Metric	Dest. Seq. No.
1	1	1	123
2	0	0	516
3	3	1	212
4	4	1	168
5	4	2	372
8	1	INF	432



PROCEDURES/SPECIFICATIONS:

The main requirements the the team utilized for implement this simulation had the following features:

1. Full routing table updates should be exchanged between neighboring nodes every advertisement cycle.
2. In the event of a tie for shortest path, the next hop in the routing table should always point to the nodeID with the lowest numerical value.
3. Set all routes in the routing table through a neighbor to the infinity value if it hasn't given any updates for some period of time.
4. Set routes in the routing table to the infinity value if they have not been updated for some period of time ("expire" them).
5. Delete routes from the routing table that have been set to infinity for some period of time ("garbage collect" them).
6. If a node or link goes down (e.g., routing daemon crashes, or link between them no longer works and drops all messages), your routing tables in the network should reflect the new network graph.

Our program followed the following specification for formatting the routing table:

- **Network ID:** for the network ID corresponding to the route, we utilized the destination IP
- **Next Hop:** the IP address of the next node in the rout/path.
- **Metric/Cost:** we utilized the cost of the route to determine the sorted path possible among the multiple paths to the same destination.
- **Interface:** We had to import an open source library in order to be able to properly use the needed functions to indicate which network interface is used.

RESULTS:

During the testing period, we were able to simulate the demanded requirements. First, we simulated the vector distance routing protocol (VDRP) using the distance vector routing algorithm with 4-5 nodes (*simulated routers*) in the network.
Then we created a routing table before storing it in the local host. Next, we ensure that the routing tables were updated periodically and process incoming advertisements from neighbors. We also set updates to the routing tables whenever there is a shortest path available in the network, without causing any communication failures.

CONCLUSION:

Throughout the developing cycle of this project, the team had some minor setbacks due to our inexperience developing a complex network application such as this. However, we were able to gather sufficient, relevant and reliable information that assisted us with much of the challenges. Fortunately, every member in the team was able to coordinate their efforts that led to the successful completion of the project, satisfying all the assigned features.