# TRADITIONAL ROUTING PROTOCOLS

## DISTANCE VECTOR ROUTING (DV)

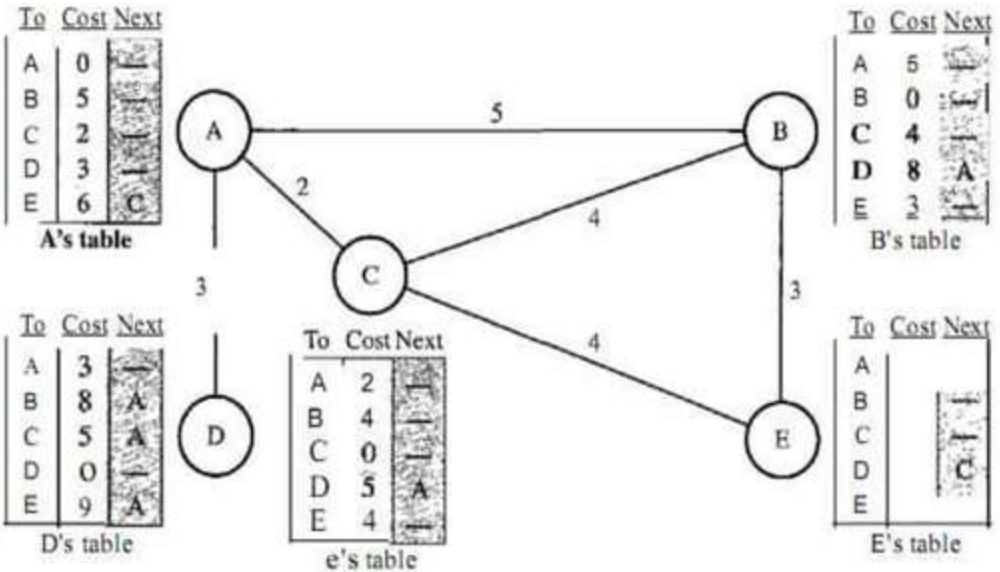
# DEFINITION

Distance is a measure of **number of Hops** the Packet requires to reach the Destination.

Here Vector is defined as (Distance, Direction) **next Hop router** to which the packet is to be forwarded.

It is also known as **Bellman Ford Algorithm**.

# EXAMPLE


To Cost Next
A 0 —
B 5 —
C 2 —
D 3 —
E 6 C
A's table
A
B
C
D
E
5
2
3
4
4
3
To Cost Next
A 5 —
B 0 —
C 4 —
D 8 A
E 3 —
B's table
To Cost Next
A 3 A
B 8 A
C 5 A
D 0 —
E 9 A
D's table
To Cost Next
A 2 —
B 4 —
C 0 —
D 5 A
E 4 —
e's table
To Cost Next
A —
B —
C —
D —
E C
E's table

# EXPLANATION

In Distance Vector Routing Protocol, each node shares its routes in the network **only to the neighbors** and does not broadcast it. (not to all)

Whenever any node receives the Routing Information it updates its own routing table and inform to its neighbors.

# WORKING PRINCIPLE

1) Firstly each node enters the cost of the neighboring node

2) A link that is down (not a direct neighbor) is assigned a cost is assigned to Infinity

# WORKING PRINCIPLE

3) Every node that sends a message directly connected to the adjacent node about the adjacent neighbors and their cost.

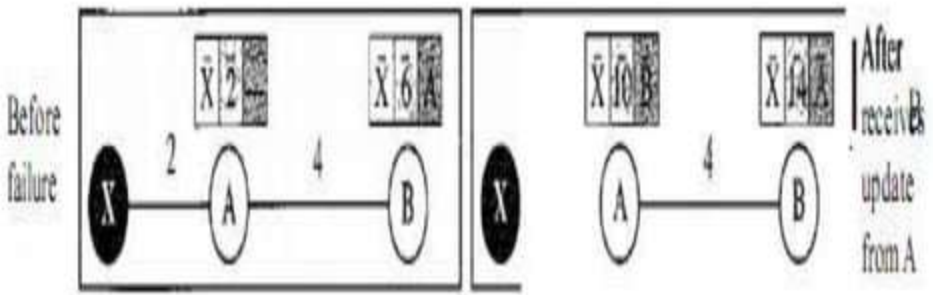
4) After exchanging the nodes information it will find the least cost to reach the other nodes information.

# DRAWBACKS OF DISTANCE VECTOR

1) COUNT TO INFINITY PROBLEM.

2) SPLIT HORIZON

# Count to infinity problem



# Split Horizon

In this case, node A keeps the value of infinity as the distance to X. Later when node A sends its routing table to B, node B also corrects its routing table. The system becomes stable after the first update.

In our scenario, node B eliminates the last line of its routing table before it sends it to A

# Split Horizon with Poison Reverse

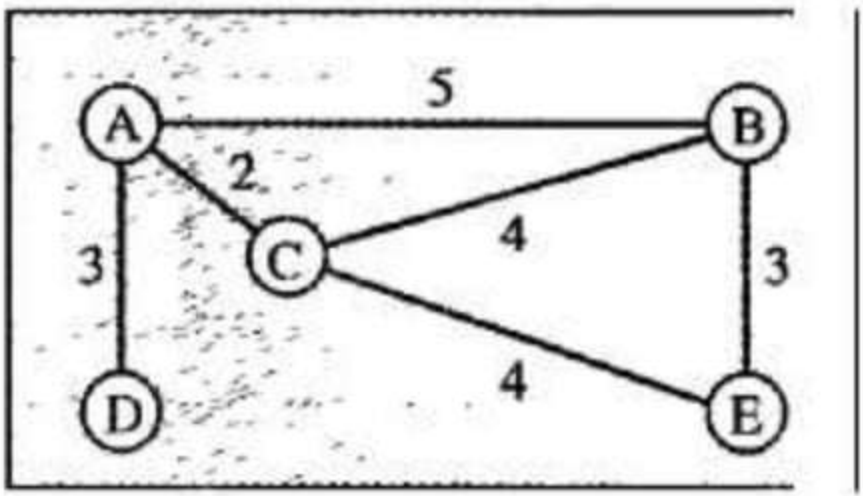
Split Horizon and Poison Reverse Using the split horizon strategy has one drawback. Node B can still advertise the value for X, but if the source of information is A, it can replace the distance with infinity as a warning: "Do not use this value; what I know about this route comes from you."

# Link State Routing

# DEFINITION

In link state routing, if each node in the domain has the entire topology of the domain the list of nodes and links, how they are connected including the type, cost (metric).

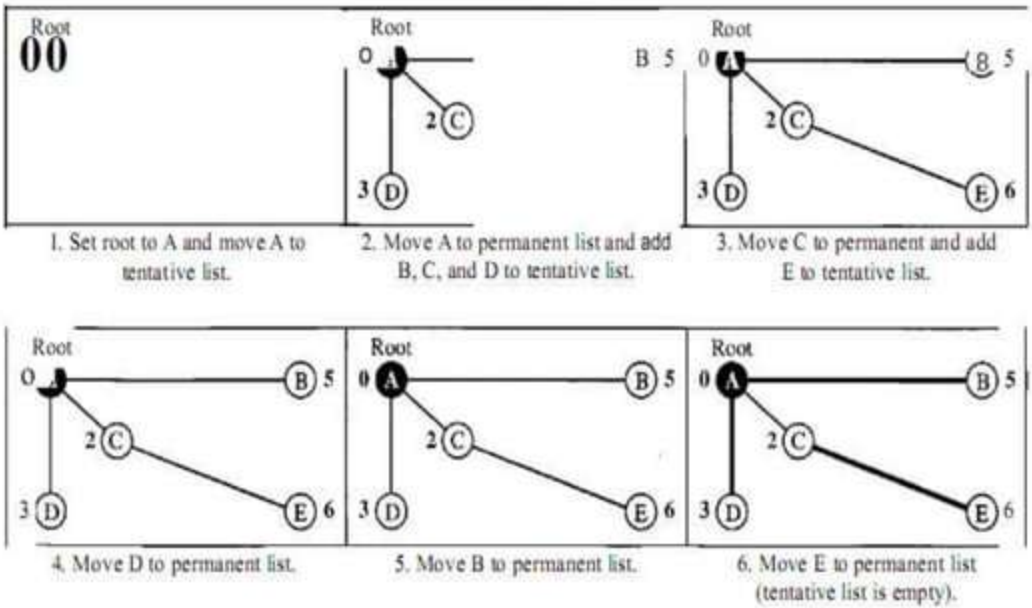
# EXAMPLE



# EXPLANATION

Node A knows that it is connected to node B with metric 5, to node C with metric 2,and to node D with metric 3. Node C knows that it is connected to node A with metric 2,to node B with metric 4, and to node E with metric 4. Node D knows that it is connected only to node A with metric 3. And so on.

# Dijkstra algorithm




1. Set root to A and move A to tentative list.

2. Move A to permanent list and add B, C, and D to tentative list.

3. Move C to permanent and add E to tentative list.

4. Move D to permanent list.

5. Move B to permanent list.

6. Move E to permanent list (tentative list is empty).

# ROUTING TABLE

| *Node* | *Cost* | *Next Router* |
|---|---|---|
| A | 0 | - |
| B | 5 | - |
| C | 2 | - |
| D | 3 | - |
| E | 6 | C |

# EXPLANATION

**Link State Advertisement** is used by all the Routers to know the cost information of all Routers participate in the network.

**Link State Packet Database** is used to store the Routing table entries.

Each node participating in the network will have identical Link State Packet Database.

# EXPLANATION

One of the Characteristics of Link State Protocol is every node constructs a graph about the connectivity of adjacent nodes.

After the graph is drawn, shortest path is calculated to reach the destination.

# PROS OF LINK STATE ROUTING

1) Scalability is not an issues in this.

2) Whenever more than 100 routers are employed link state routing is used.

# CONS OF LINK STATE ROUTING

1) It contain more CPU utilization

2) It is more difficult to configure and maintain
( because more number of routers are going to be employed)